LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-CONF-404117

Accomplishments and Challenges in Code Development for Parallel and Multimechanics Simulations

Tony Degroot, Robert Ferencz, Mark Havstad, Neil Hodge, Dennis Parsons, Michael Puso, Jerome Solberg, Ed Zywicz

May 23, 2008

VECPAR'08 # 8th International Meeting High Performance Computing for Computational Science
Toulouse, France
June 24, 2008 through June 27, 2008

Accomplishments and Challenges in Code Development for Parallel and Multi-mechanics Simulations

Michael Puso, Tony Degroot, Robert Ferencz, Jerry Lin, Dennis Parsons, Jerome Solberg, Ed Zywicz

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344

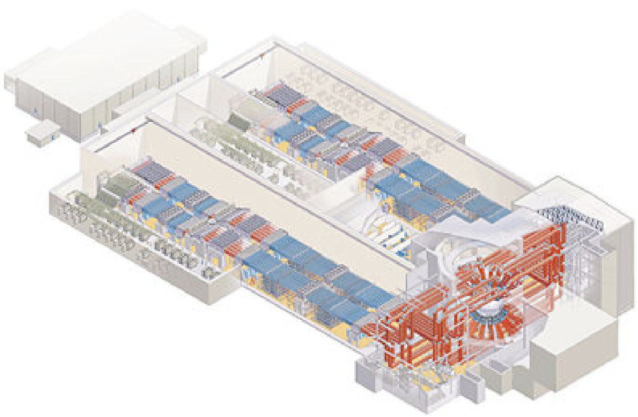
Methods Development Group at Lawrence Livermore National Laboratory

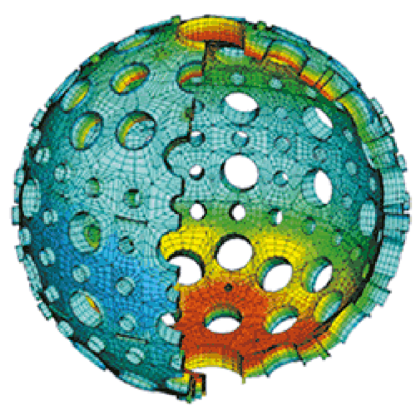
- Lawrence Livermore National Laboratory (LLNL)
  - Located 40 Miles (64 Km) East of San Francisco, California
  - A R&D institution for science and technology applied to national security.
  - Lots of parallel multiphysics codes
  - Lots of big computer hardware; *9 in the top 500*

1.	Blue Gene/L	IBM	212,992 CPU's	478 Tflops R_{max}
11.	Purple	IBM	12,288 CPU's	76 Tflops R_{max}

29. Atlas; 38. Minos; 47. Thunder; 61. Rhea; 241. Zeus; 414. ALC; 449. Lilac;

  - Home of National Ignition Facility (NIF)
    - Worlds largest laser, 192 Beams, 1.8 Megajoule per shot (same Laser MegaJoule)





Methods Development Group at Lawrence Livermore National Laboratory

ENGINEERING

- Lawrence Livermore National Laboratory (LLNL)
  - 7000 employees; "Critical Mass" ;-) of expertise

- Methods Development Group
  - Works closely with LLNL Engineers
  - Supports Los Alamos and Select Department of Defense Sites
  - Organized in 1975 => *DYNA3D* code
    - *DYNA3D:* first full capability non-linear explicit finite element code
    - Originally developed by John Hallquist
      - 1989 starts LSTC to develop commercial LS-DYNA
      - Predecessor to PAM-CRASH

Methods Development Group at Lawrence Livermore National Laboratory

ENGINEERING

- Methods Development Group
  - Later developed
    - *NIKE3D* (Implicit non-linear structural f.e)
    - *TOPAZ3D* (Thermal f.e.)
  - Current code developments
    - *PARADYN* (Massively parallel version of DYNA3D)
      - Originally developed by Carol Hoover and Tony Degroot ~1996
      - Couplings:
        - Structural, Thermal and Fluid Structural (DYSMAS)
    - *DIABLO* (Massively parallel implicit FE code)
      - Originally developed by Bob Ferencz ~2001
      - Couplings:
        - Structural, Thermal, Advection-Diffusion, Electromagnetics
- Other LLNL code developments
  - *ALE3D, EMSolve, MERCURY etc.*

Outline:

- Transient Explicit Code: *PARADYN*
  - Applications
  - Formulation
    - Parallel Contact Algorithms: Node-on-Segment w/ evolving surfaces
  - Meshless (or Meshfree)
  - Fluid-Structure Interaction
    - Embedded mesh coupling w/ GEMINI (Indian Head Naval Base)
- Low Frequency/Statics Implicit Code: *DIABLO*
  - Formulation
    - Parallel Contact Algorithms: Segment-to-Segment (mortar)
    - Solution/Coupling strategy
  - Applications
- Future Work

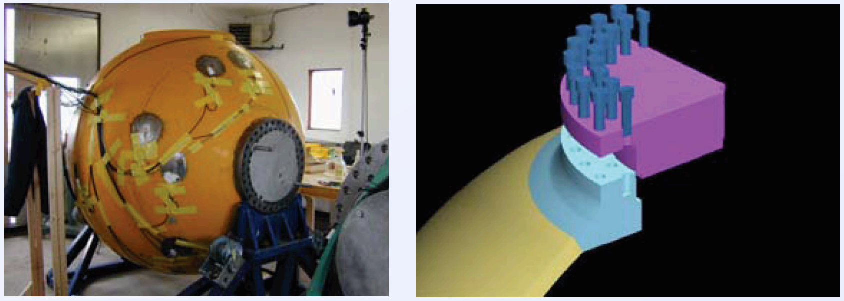
Outline: Issues in Parallel/Coupled Code Development

- Major Development Areas: (no order)
  - Contact:
    - Solid Mechanics, Thermal Mechanics, Electromagnetics
  - Discretization Methods:
    - Finite Elements (better for Solids, Electromagnetics)
    - Finite Volume (better for Fluids?)
    - Lagrangian /Eulerian/ ALE?
    - Meshless (better for material failure?)
    - Embedded Mesh (Overlapping meshes)
  - Solution Strategies:
    - Time Integration: Explicit vs Implicit
    - Non-linear schemes, Parallel linear solvers
    - Mechanics coupling schemes (Partitioned vs. Monolitic)
    - Adaptivity
  - Material Models:
    - Solid Mechanics (Material Failure/Fracture/Multiscale)

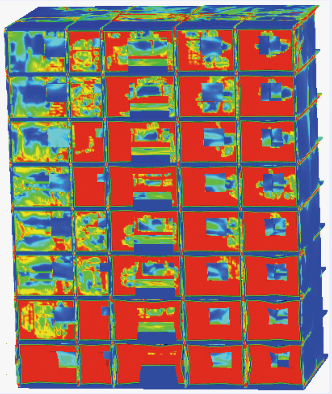
Transient Dynamic Analysis: Explicit Time Integration

- Transient Dynamic Analysis:
  - High rate loadings typically on the order of microseconds
  - Simulate; Drop tests, Pressure Vessels, Blast Loadings, Crashes, etc.

Transportation container flange detail (Courtesy of Dan Badders, LLNL).



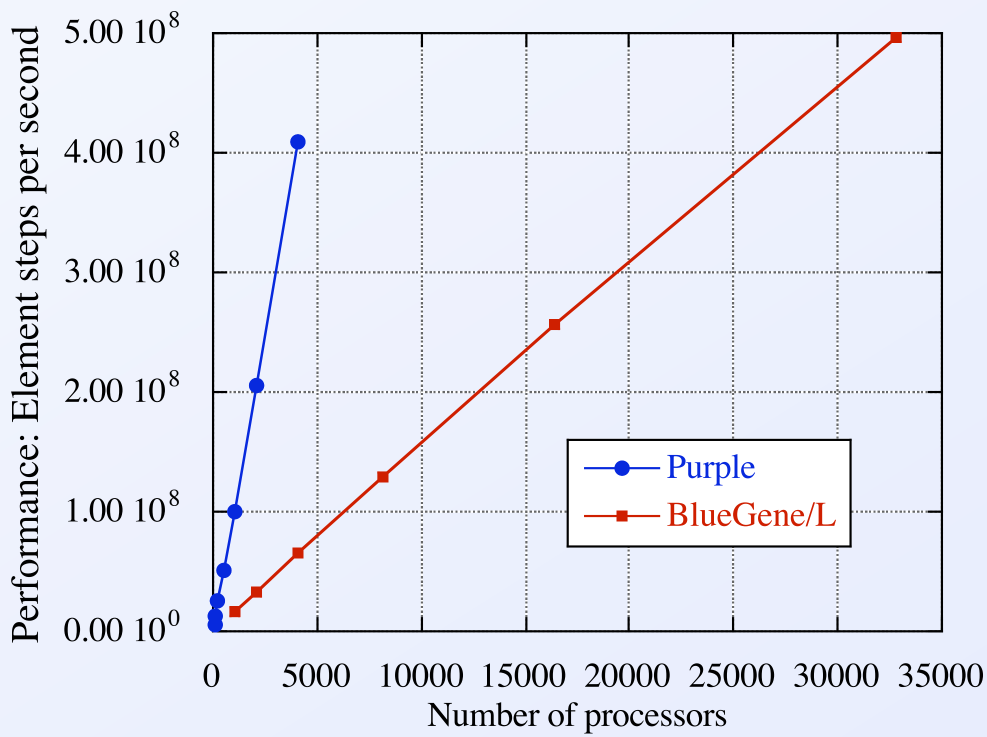
Hydrodynamic containment vessel (LANL Weapons Engineering).



Blast loading on apartment building.
30 million degrees of freedom
(P. Papados, U.S. Army ERDC)

Transient Dynamic Analysis: Explicit Time Integration

- Simple to parallelize
  - Build static decomposition using METIS (Karypis and Kumar, 1998)
  - Scales great; consider 90 million element problem

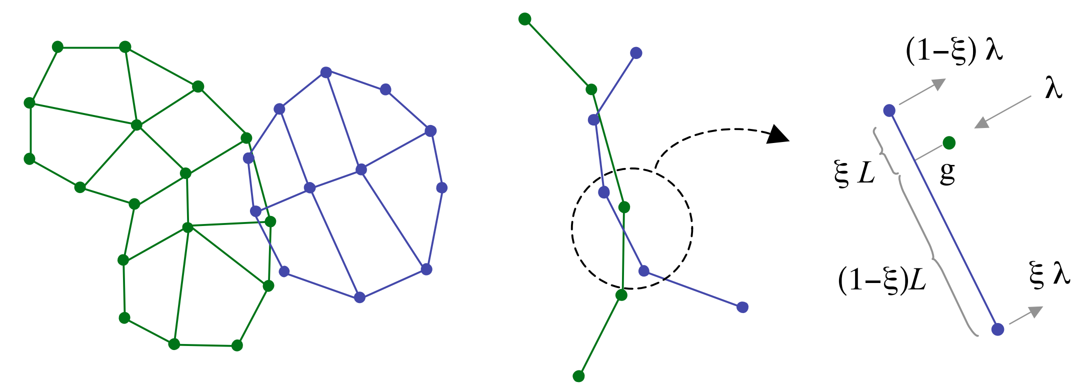


Explicit Time Integration w/ Node-on-Segment Contact

- Explicit time integration
  - Conditionally stable Δt = *h*/*c*
  - Equations of motion

$$Ma_{n+1} + f_n^{\text{int}}(x_n) + f^c = f_n^{ext}$$

M – diagonal mass matrix
f - force vector (internal, contact and external)
x_n, v_n, a_n nodal position, velocity and acceleration vectors at time t_n



$$f^c = \begin{Bmatrix} 1 \\ \xi \\ (1-\xi) \end{Bmatrix} \lambda$$

Explicit Time Integration w/ Node-on-Segment Contact

- Solve for acceleration

$$a_{n+1} = M^{-1}(f_n^{ext} + f_n^{\text{int}}(x_n) + f^c) \qquad \text{with penalty method} \quad \lambda = \kappa\, g_n$$

- Update velocities and positions

$$v_{n+1/2} = v_{n-1/2} + \Delta t_n a_{n+1} \qquad x_{n+1} = x_n + \Delta t_n v_{n+1/2}$$

- Penalty method issues; is κ sufficiently large? too large $\Rightarrow$ unstable
- Lagrange multiplier method forces $g_{n+1} = 0 \;\Rightarrow \lambda$ unkown
- Using predictor-corrector method (Zywicz and Puso, IJNME, 1998)

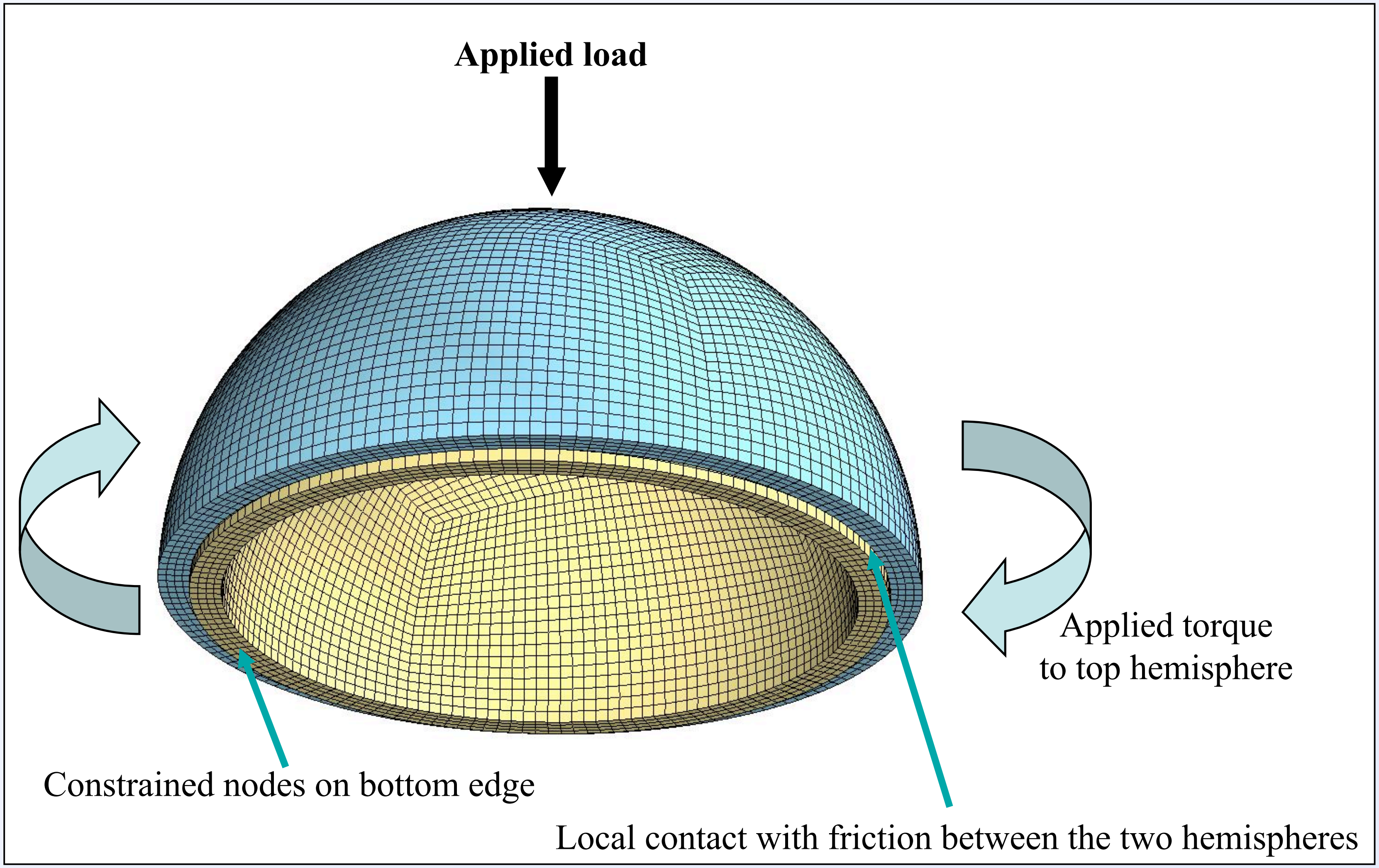
$$g_{n+1} = g'_{n+1} - \Delta t^2 G_{n+1}^{\;t}\; M^{-1} G_{n+1}\; \lambda_{n+1} \qquad f^c = G_{n+1}\, \lambda_{n+1}$$

- Satisfy Kuhn-Tucker constraints

$$\lambda_{n+1} \geq 0, \quad g_{n+1} \geq 0, \quad g_{n+1}\, \lambda_{n+1} \geq 0$$

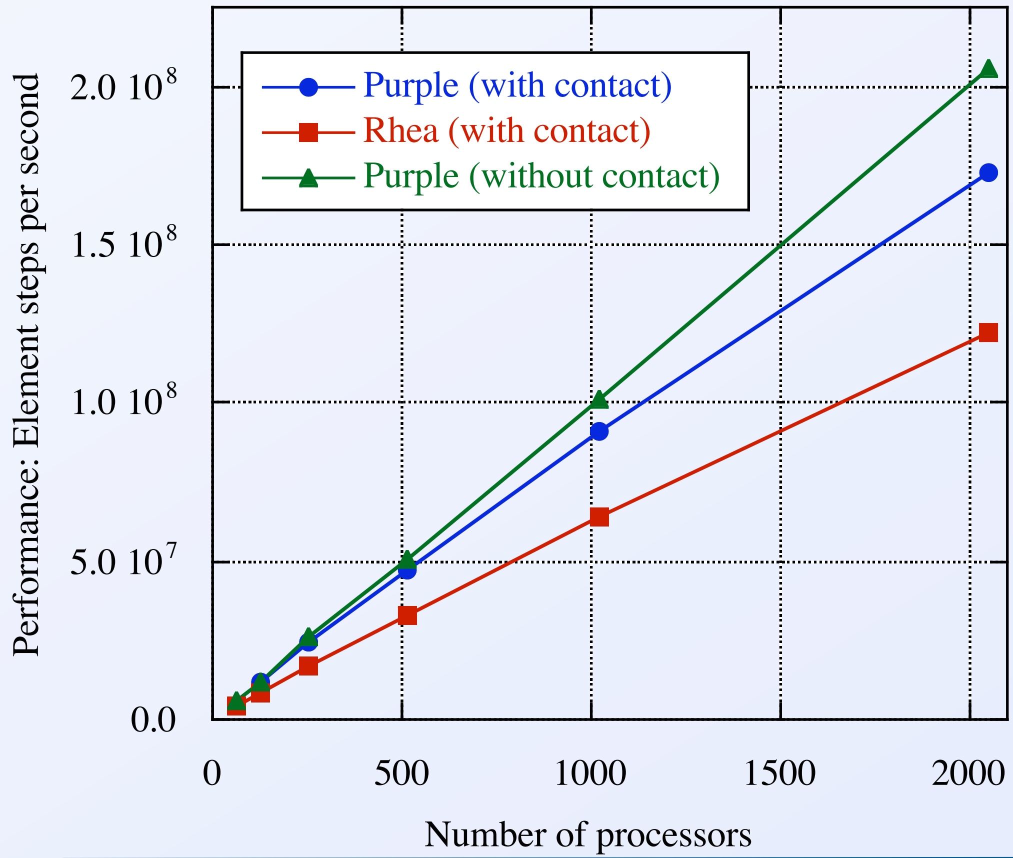
- Use constrained preconditioned conjugate gradient method
  - Solve matrix free
  - Do dynamic partitioning using METIS
  - Sometimes need to eliminate redundant constraints

Timing study: two nested hemispheres (Juicer)



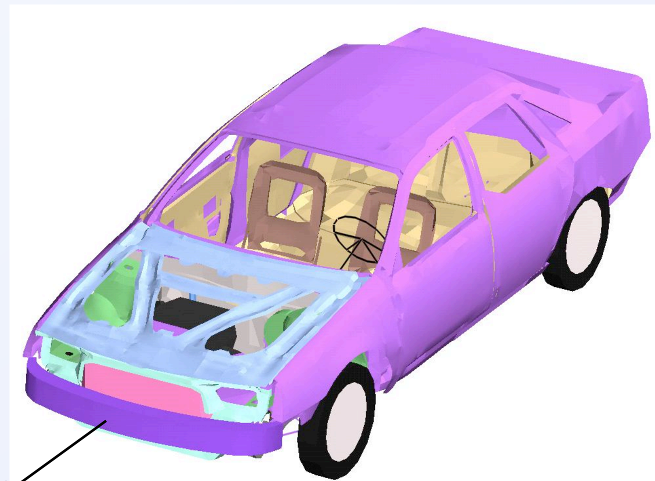
Timing study: Juicer

- 9 million elements
  - Scales well
  - Cost 30% more than problem without contact

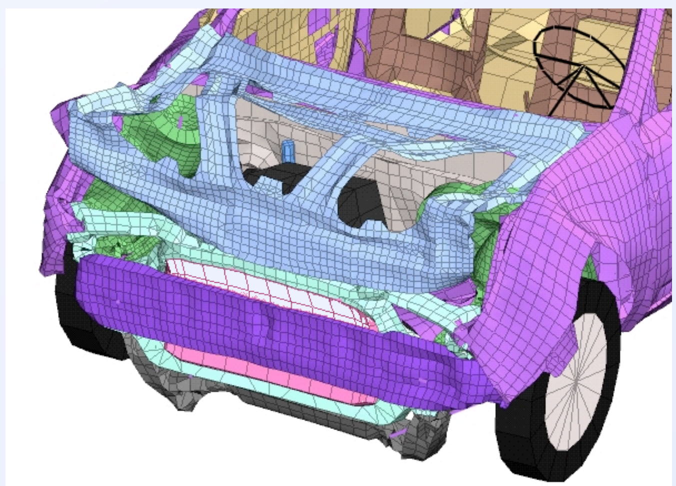


Lagrange Multipliers Work Great!

- Use on most of our problems
  - Costs 10-50% more
  - e.g., Ford Taurus



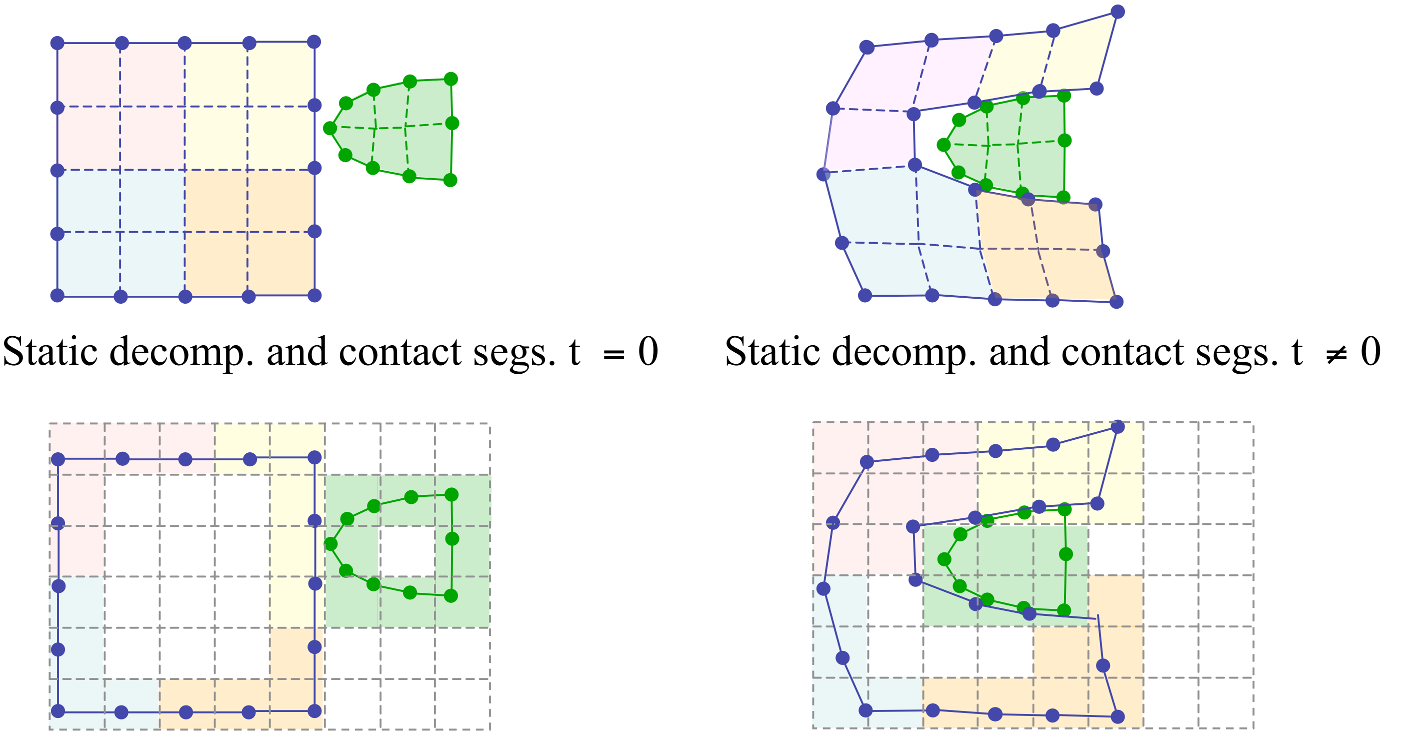
35 MPH



Frontal Impact with Rigid Barrier
courtesy of Ed Zywicz

Parallel Contact Search

- Automatic Contact
  - Contact nodes and segments computed automatically
  - Element erosion due to damage creates new contact segments
  - *Use Dynamic Decomposition for contact*

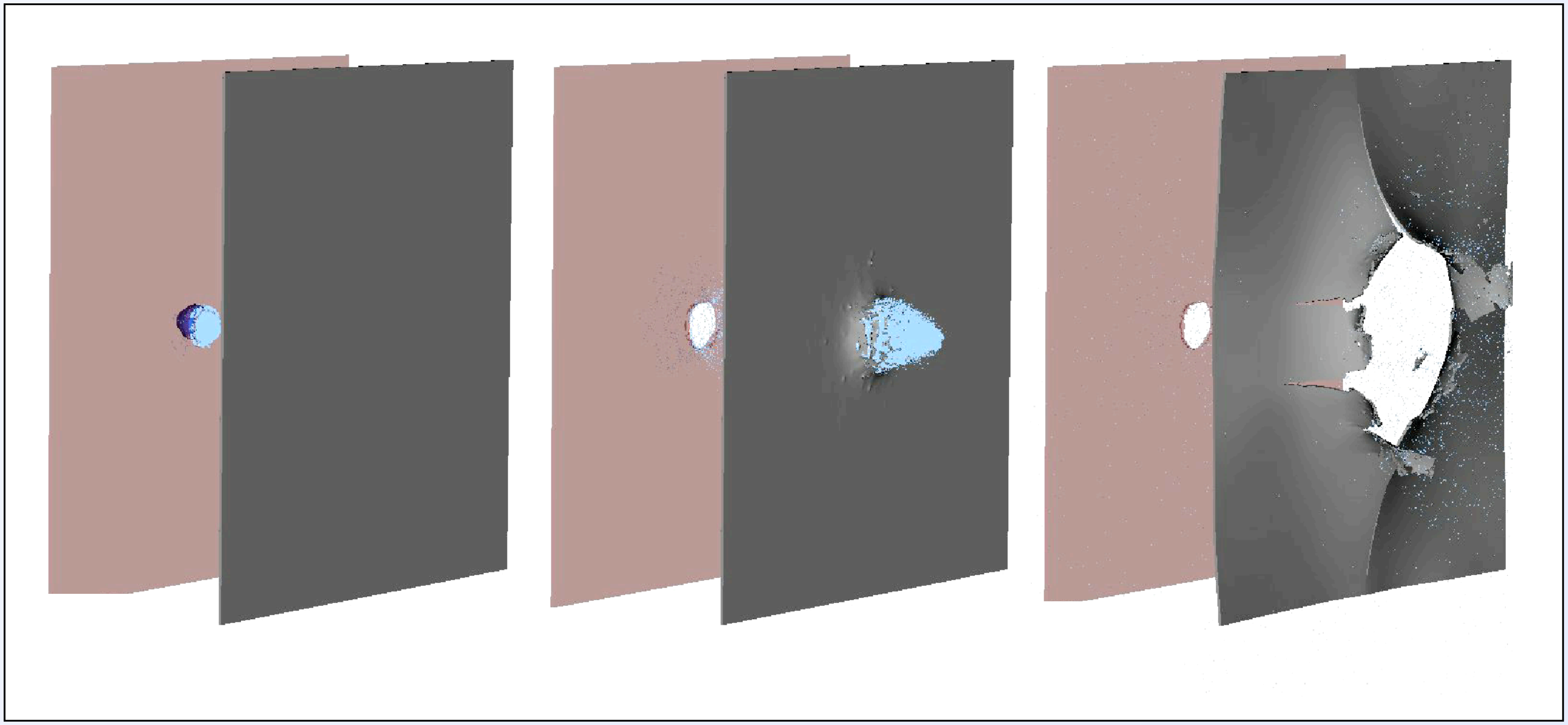


Static decomp. and contact segs. $t = 0$ | Static decomp. and contact segs. $t \neq 0$

Dynamic contact decomp. segs. $t = 0$ | Dynamic contact decomp. segs. $t \neq 0$

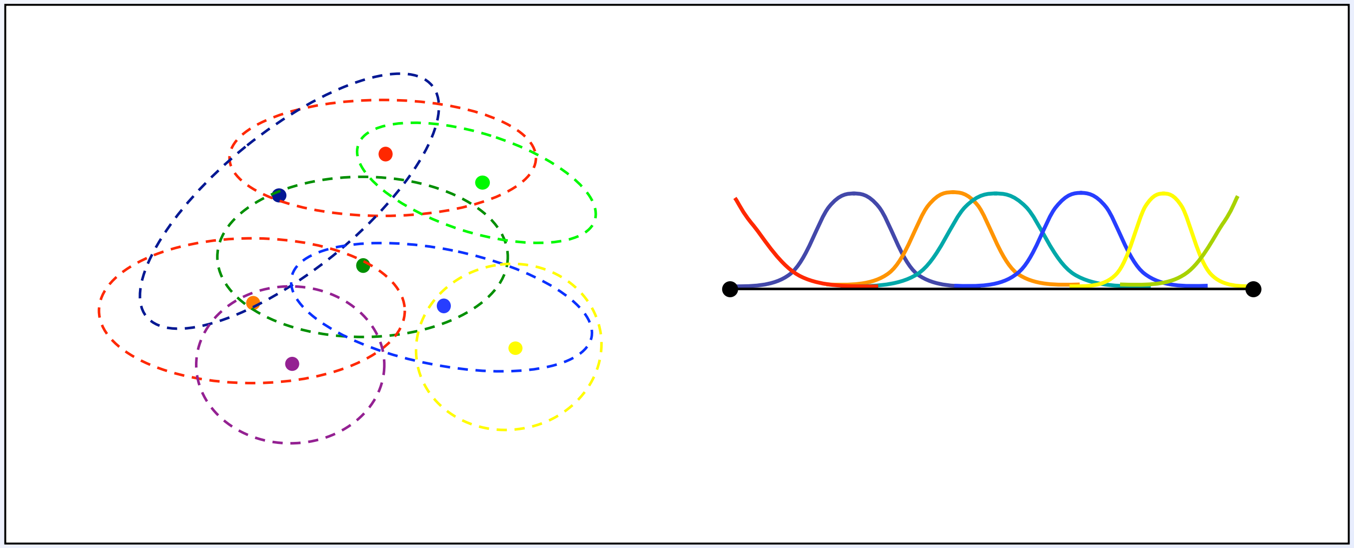
Parallel Contact Search

- Consider penetration of parallel plates
  - Elements erode and node become “mass particles” to avoid tangling



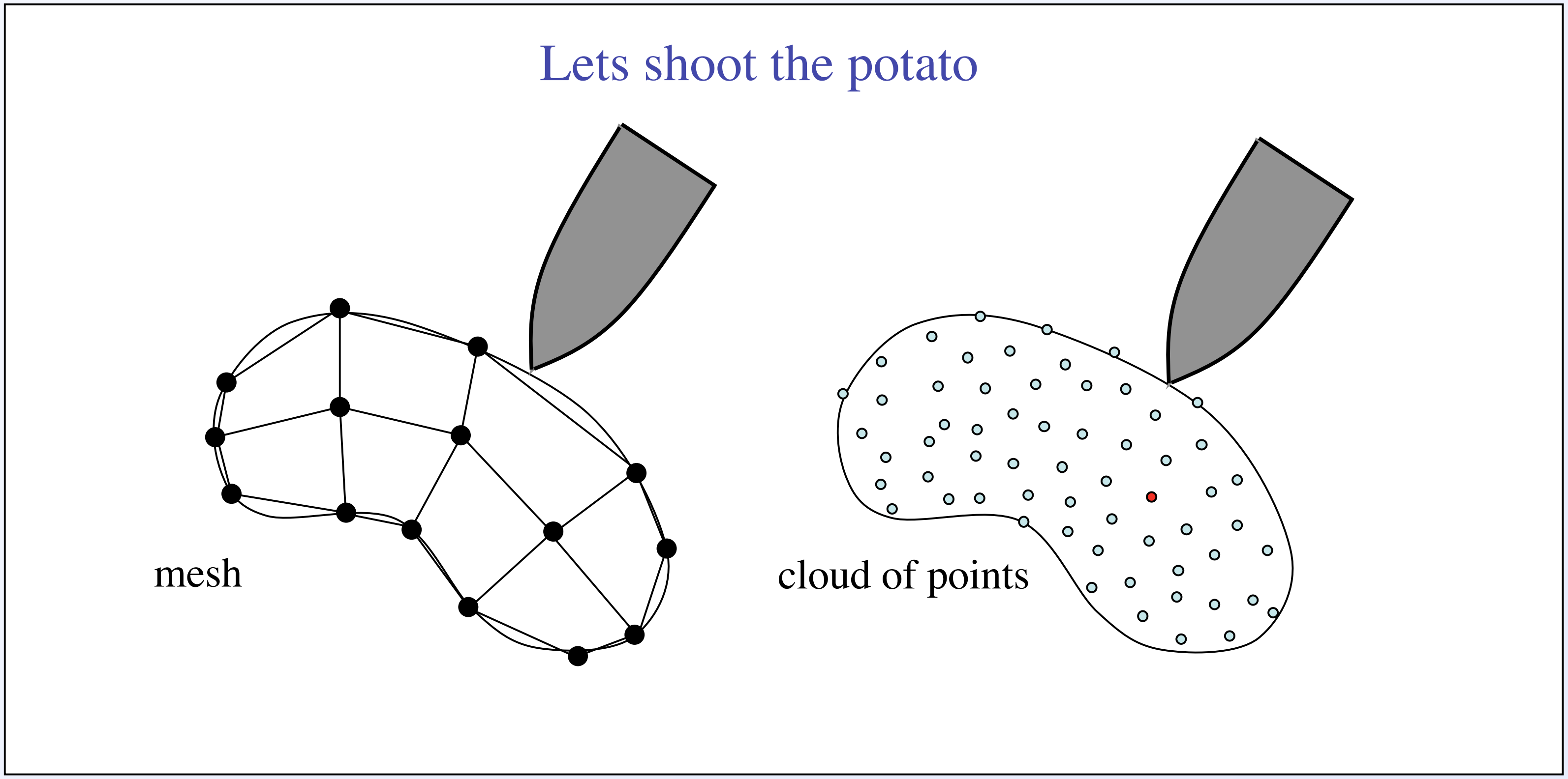
Meshfree (formerly known as Meshless)

- Meshfree avoids problems with mesh tangling
  - Use shape functions based on proximity of neighbors
  - Shape function supports typically spherical (or elliptical)
    - e.g. Moving Least Squares (Belytschko et.al 1993)
    - Produce linear exact shape functions on a cloud of points
    - Parallelization similar to “Automatic Contact”



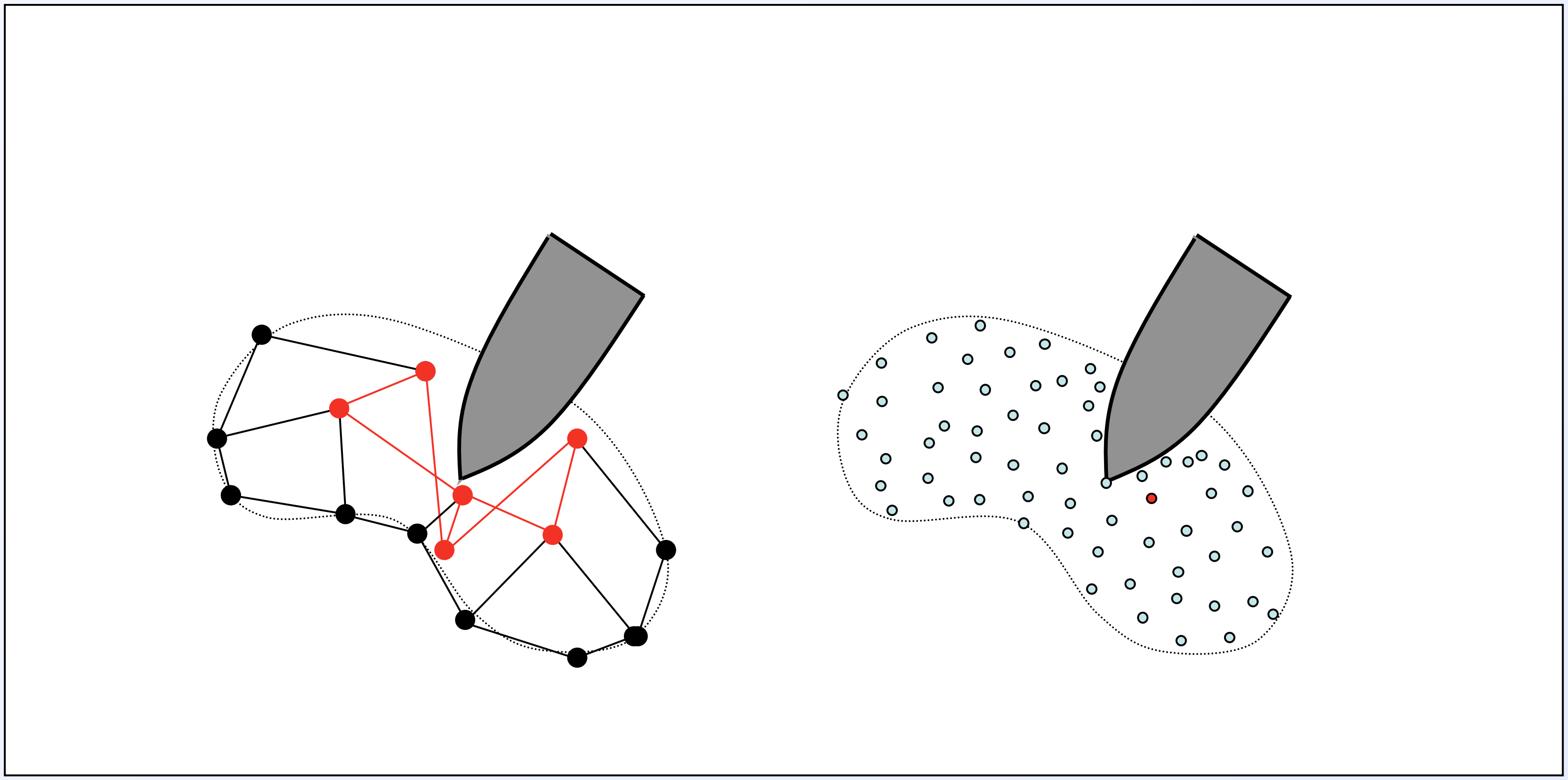
Meshfree: Mesh tangling

- Meshfree avoids problems with mesh tangling



Meshfree: Mesh tangling

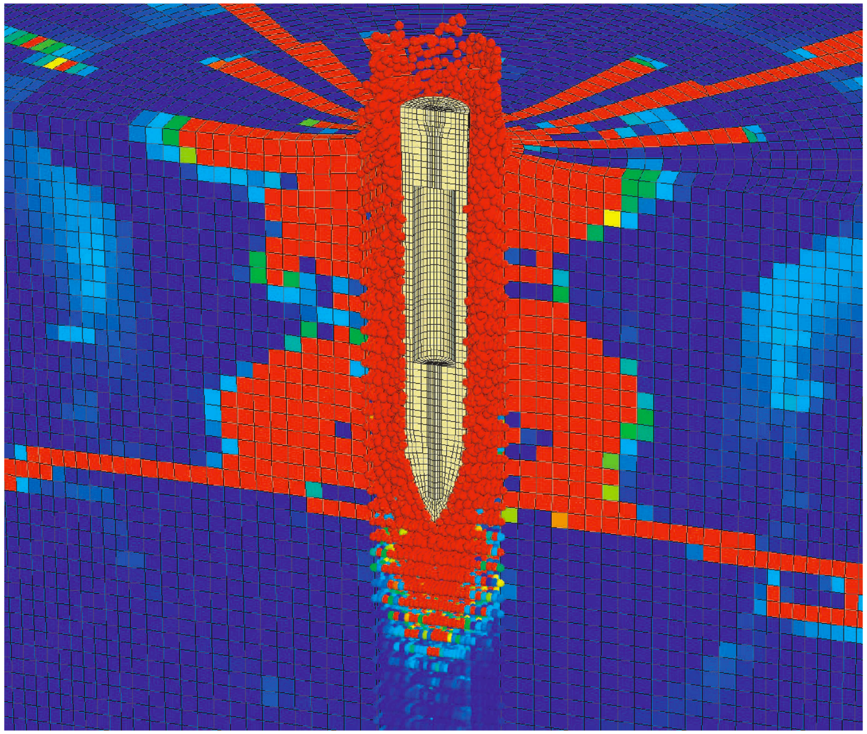
- Meshfree avoids problems with mesh tangling
  - Connectivity defined "on the fly"

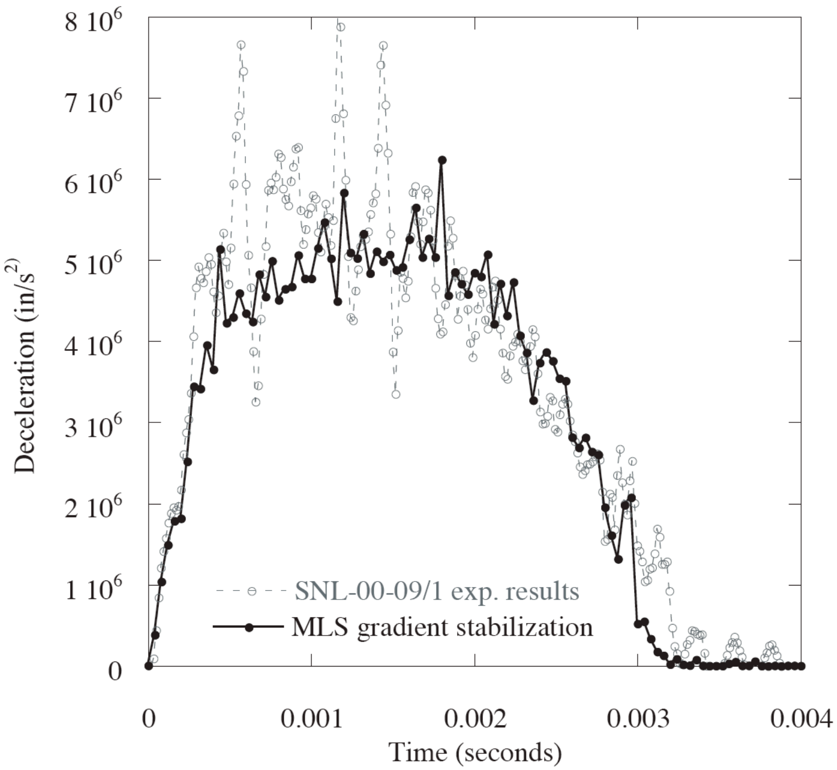


Meshfree: Validation

- Compare to simulation to penetrator experimental results (Puso, Chen, Zywicz, Elmer; 2008)

M.J. Forrestal, D.J. Frew, J.P. Hickerson, and T.A. Rohwer. Penetration of concrete targets with deceleration-time measurements. *International Journalof Impact Engineering*, 28:479–497, 2003.

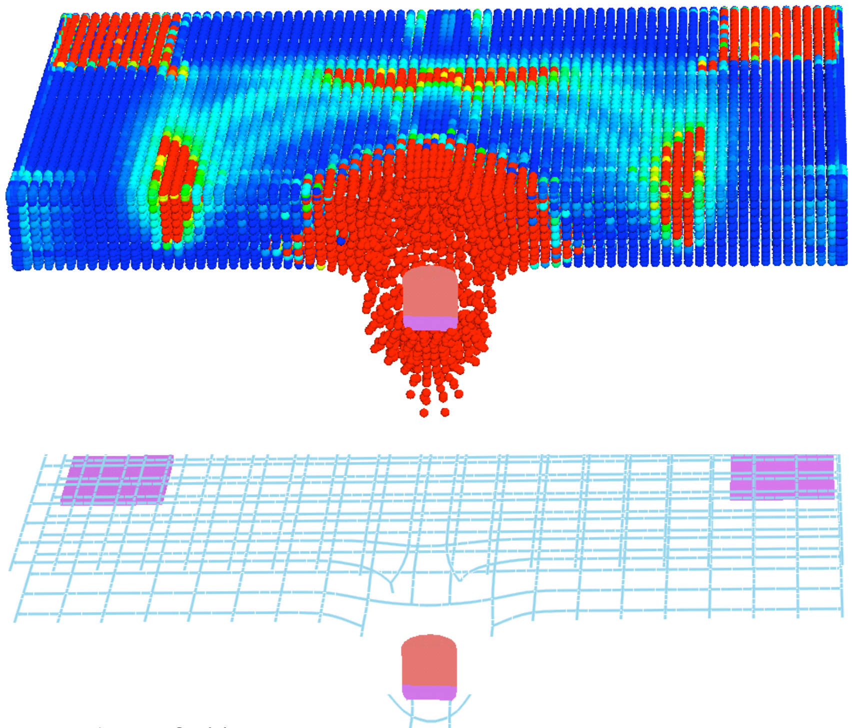




Meshfree: Validation

- Compare to simulation to penetrator experimental results
  - Compared spall profile experiment 660-880 mm vs. simulation 600 mm.

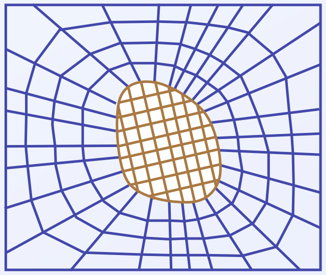
T. Sugano et. al. Local damage to reinforced concrete structures caused by impact of aircraft engine missiles Part 1. Test program, method and results. *Nuclear Engineering and Design*, 140:387-405, 1993.



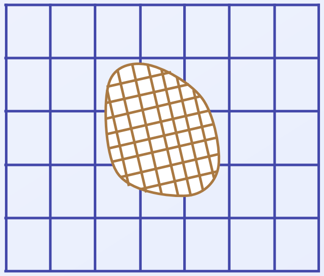
Rebar failure

Fluid Structure Interaction

- PARADYN (parallel structural FE) coupled to GEMINI (parallel fluid FV)
  - DYSMAS (Wardlaw, Ludon, Renzi, Kiddy, McKeon; 2003)
  - Partitioned coupling made through embedded mesh
- What is an embedded mesh
  - Avoids body fitted mesh
  - For example, superpose solid body onto fluid grid



Body fitted mesh



Embedded mesh

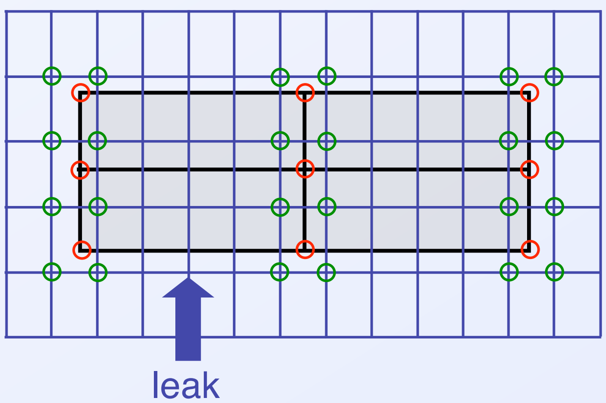
Embedded mesh techniques

- Existing *Embedded Mesh* methods for *moving meshes*
  - *Immersed boundary methods* (C.S. Peskin 1977, 2002)
  - *Immersed finite element methods* (W.K. Liu 2004)
  - *Overset grid methods* (W.D. Henshaw 2006)
  - *Mortar fictitious domain methods* (Baaijens 2001)

Embedded mesh techniques

- Existing *Embedded Mesh* methods for *moving meshes*
  - *Immersed boundary methods*
  - *Immersed finite element methods*
    - Enforces constraints “point-wise” between solid mesh fluid mesh
    - $h_s << h_f$ otherwise leaks i.e. not consistent

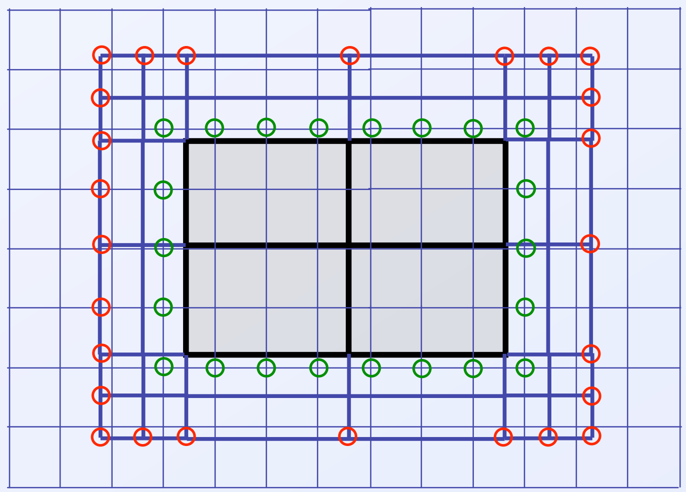
e.g. $h_s > h_f$



ENGINEERING

Embedded mesh techniques

- Existing *Embedded Mesh* methods for *moving meshes*
  - *Overset grid methods*
    - Momentum/Flux not conserved across boundary
    - Loose symmetry of [K] where [K]$u = f$
    - No Leaks, Easy

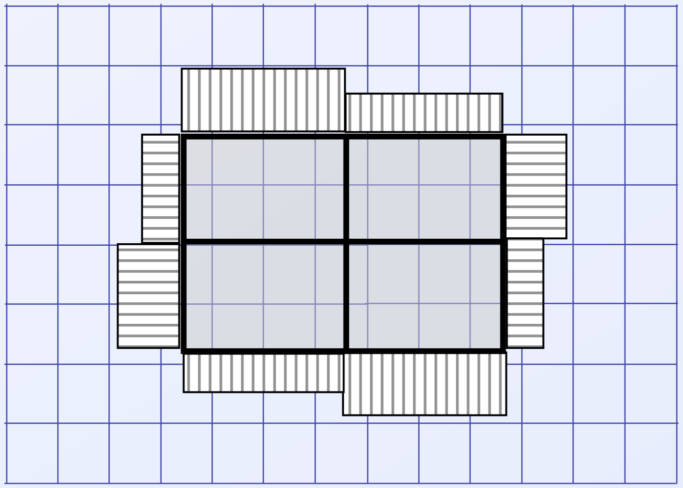


Embedded mesh techniques

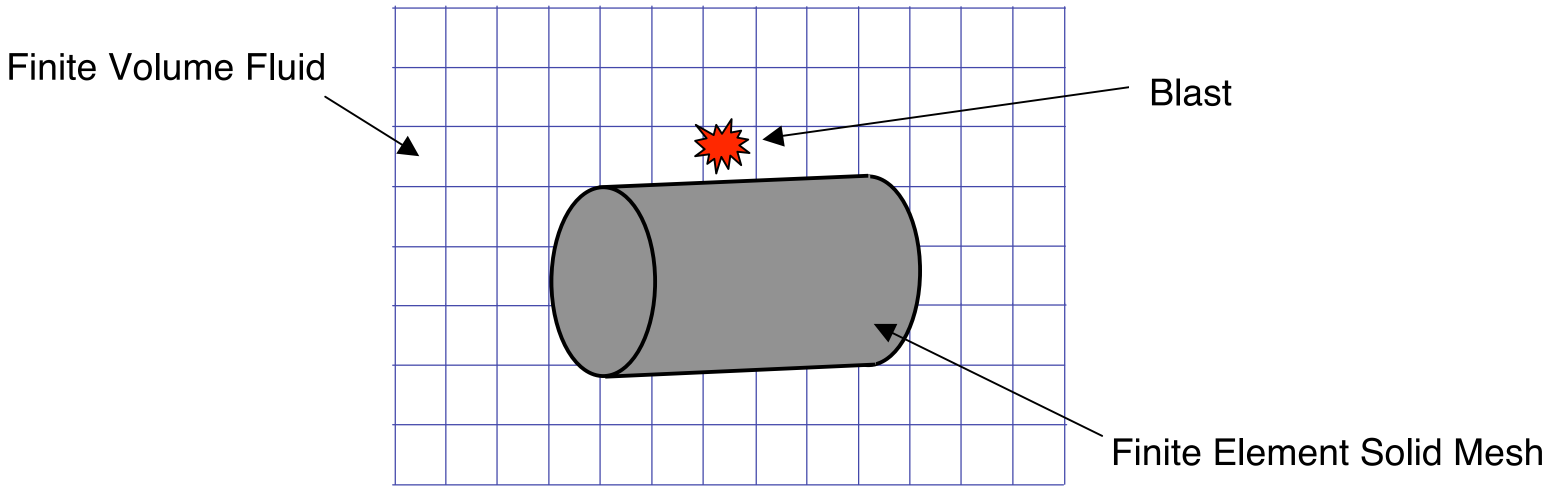
- Existing *Embedded Mesh* methods for *moving meshes*
  - *Mortar fictitious domain methods*
    - Requires surface integral
    - No Leaks, Conserves momentum, Retains symmetry

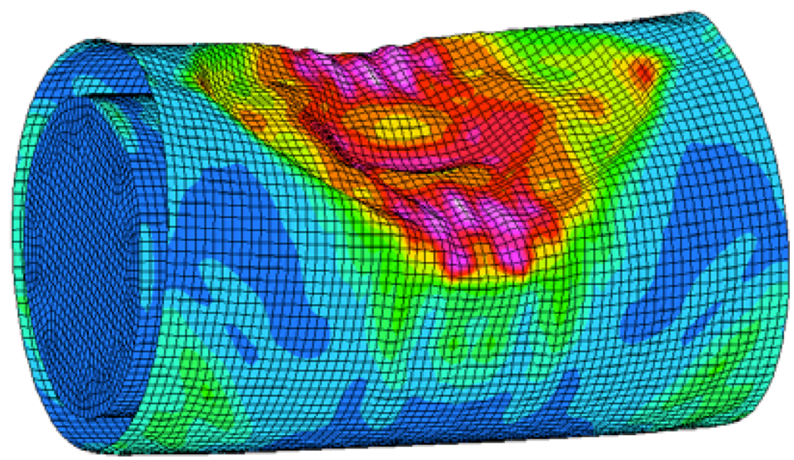
Apply surface integral to constrain fluid and solid surface velocities

$$\int_\Gamma \boldsymbol{\lambda} \cdot (\boldsymbol{v}^s - \boldsymbol{v}^f)\, d\Gamma = 0$$



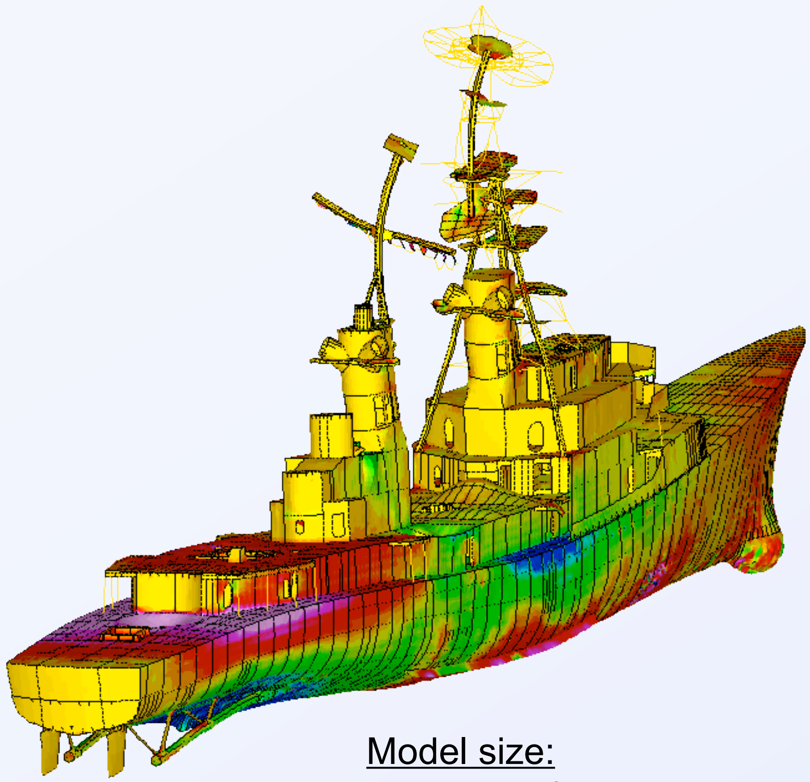
DYSMAS validation: Double Walled Cylinder





DYSMAS Validation: Ship Response to Underwater Explosion

Courtesy of Alan Luton



Model size:
300k shell & beam elements

Contours of longitudinal stress component
(Displacements magnified)

Static / Low Fequency Dynamic Analysis: Implicit Time Integration

- Implicit time integration used for problems on the order of second-years
  - Consider statics (i.e. no inertia)

$$f^{\text{int}}(x_{n+1}) + f^{c}(x_{n+1}) - f_{n+1}{}^{ext} = 0$$

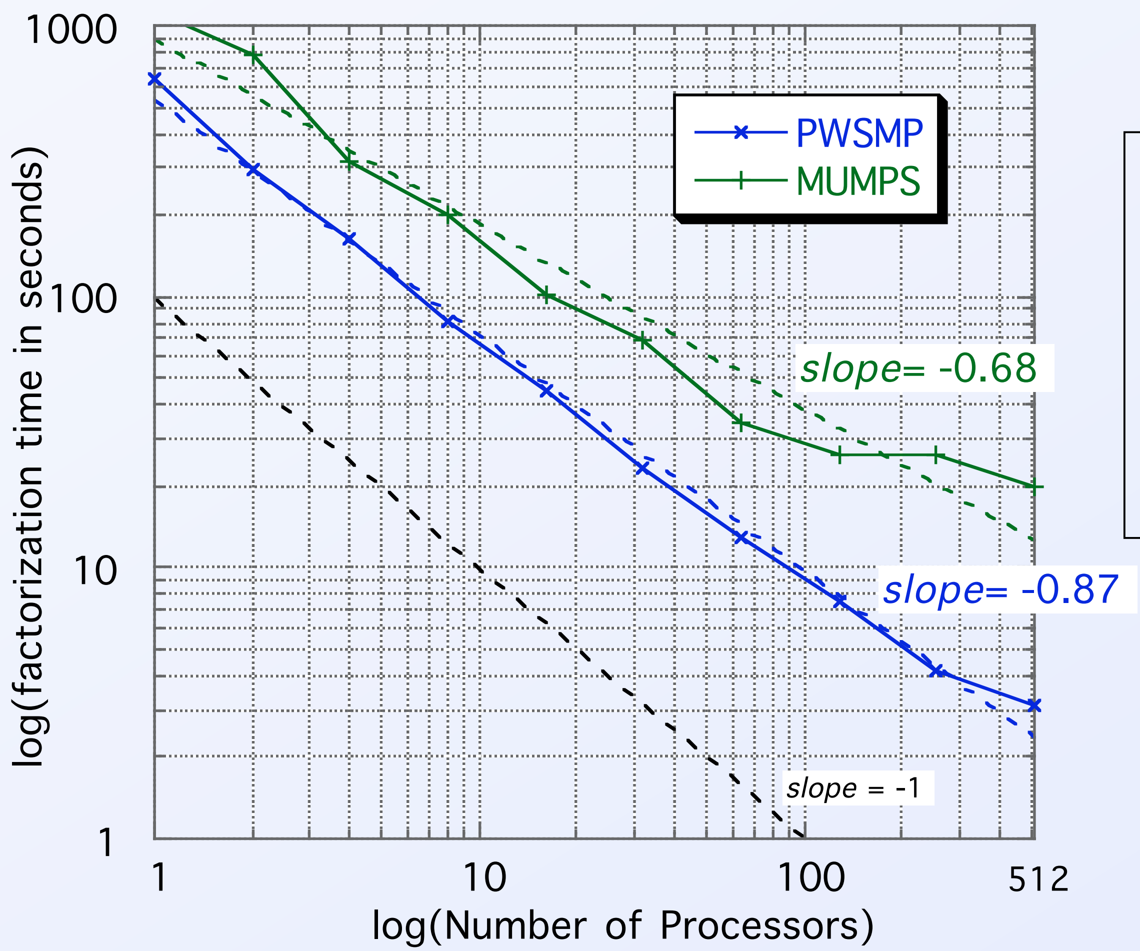
  - Now linearize i.e. Newton Raphson

$$K\, u_{n+1}^{i+1} = -f^{\text{int}}(x_{n+1}^{i}) - f^{c}(x_{n+1}^{i}) + f_{n+1}{}^{ext} \quad \text{where} \quad K = \frac{\partial f}{\partial x} \quad \text{and} \quad x_{n+1}^{i+1} = x_{n+1}^{i} + u_{n+1}^{i+1}$$

  - Must solve linear system of equations
    - Use Parallel Algebraic Multigrid Solver BoomerAMG
      - Developed at LLNL by Rob Falgout and team
      - Available for download as part of *HYPRE* library
    - Use Parallel multi-frontal direct solvers
      - PWSMP Anshul Gupta from IBM Watson Center
        - » Not freely available (lease it $5000/year)
      - MUMPS
        - » Freely available

Performance of PWSMP and MUMPS

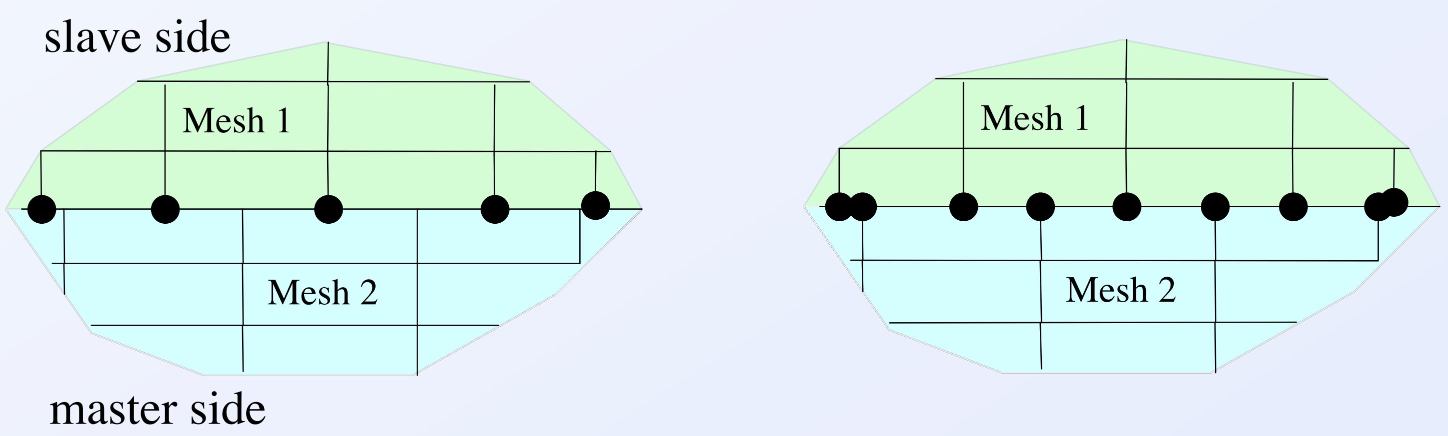
- 48x48x48 brick mesh; 345,744 degrees of freedom



Wall time	PWSMP	MUMPS
1.0000	639.40	1115.0
2.0000	295.70	789.00
4.0000	163.60	319.00
8.0000	81.750	199.20
16.000	44.850	102.00
32.000	23.280	69.010
64.000	12.980	34.110
128.00	7.4720	26.410
256.00	4.2090	26.190
512.00	3.1000	20.100

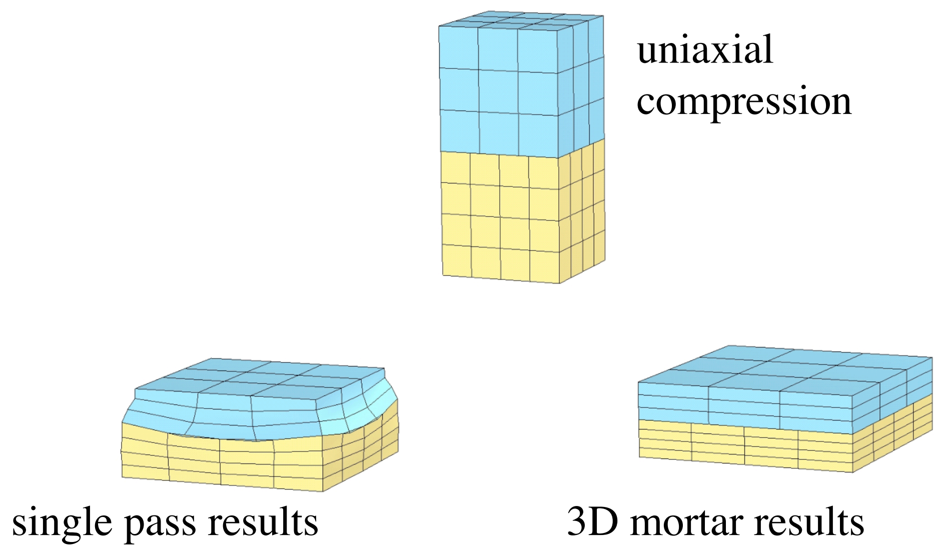
Contact Issues

- Most FE codes still use Node-on-Segment contact
- Two Types: Single Pass and Double Pass
  - Why? Because its easy!
- A number of pathologies exist for both
  - Particularly for structural mechanics and electromagnetics



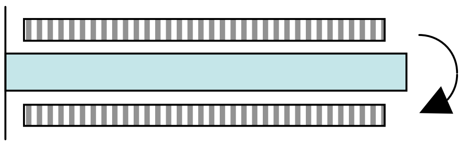
Node-on-Segment Contact Issues

- Four issues affecting robustness of *node-on-segment*
  1. Doesn't satisfy patch test (esp. single pass)

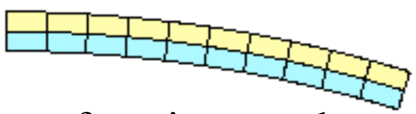


Node-on-Segment Contact Issues

- Four issues affecting robustness of *node-on-segment*
  1. Doesn't satisfy patch test (esp. single pass)
  2. Locking / overconstraint (double pass)



beam with pressure and end moment

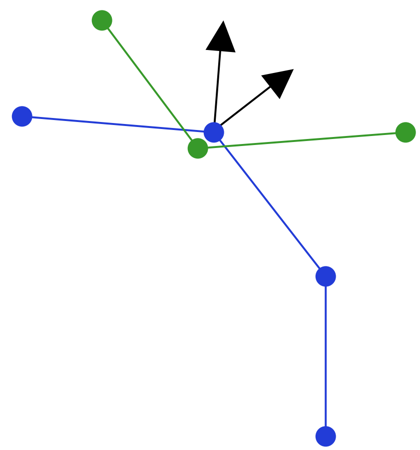


conforming mesh results

dissimilar mesh results

Node-on-Segment Contact Issues

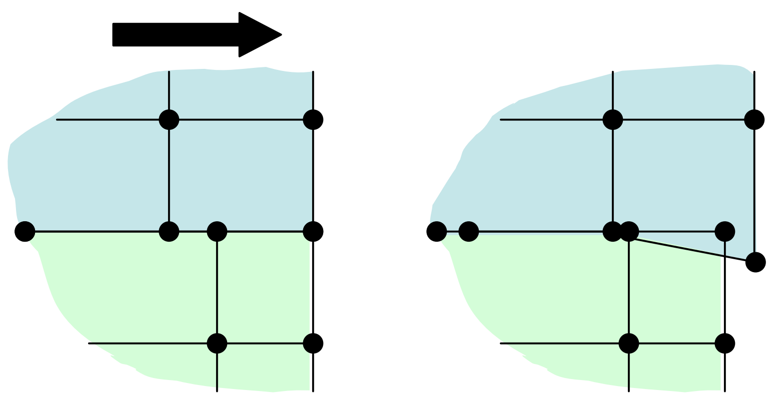
- Four issues affecting robustness of *node-on-segment*
  1. Doesn't satisfy patch test (esp. single pass)
  2. Locking / overconstraint (double pass)
  3. Non-smooth surfaces cause force jumps when sliding



normals are ambiguous

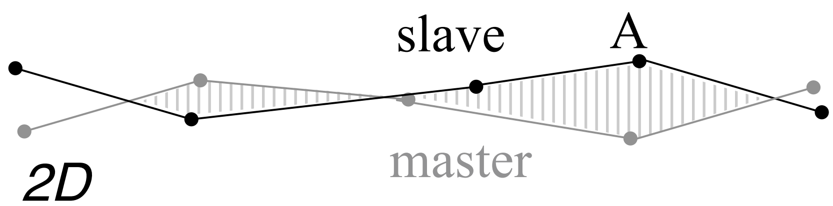
Node-on-Segment Contact Issues

- Four issues affecting robustness of *node-on-segment*
  1. Doesn't satisfy patch test (esp. single pass)
  2. Locking / overconstraint (double pass)
  3. Non-smooth surfaces cause force jumps when sliding
  4. Get jumps when slave nodes slide off boundaries



Solution: Mortar Contact

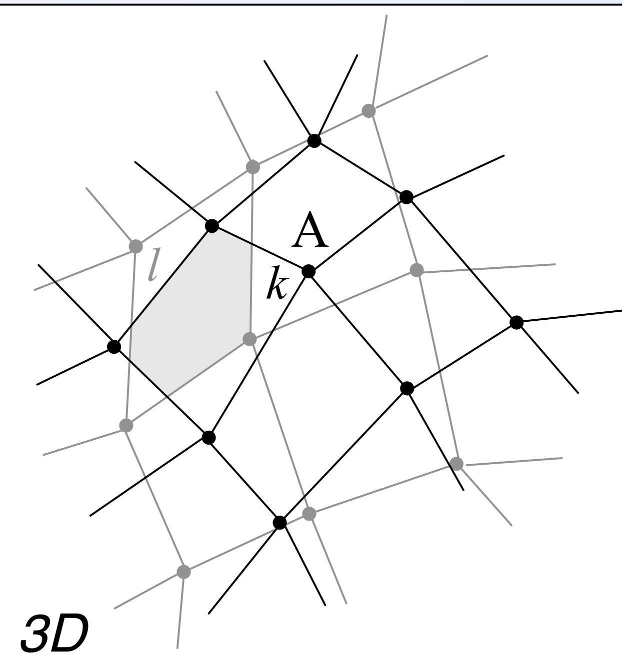
- Mortar domain decomposition developed by (Benardi, Maday, Patera; 1992)
- Extended to general 3D large deformation frictional contact (Puso, Laursen; 2004)



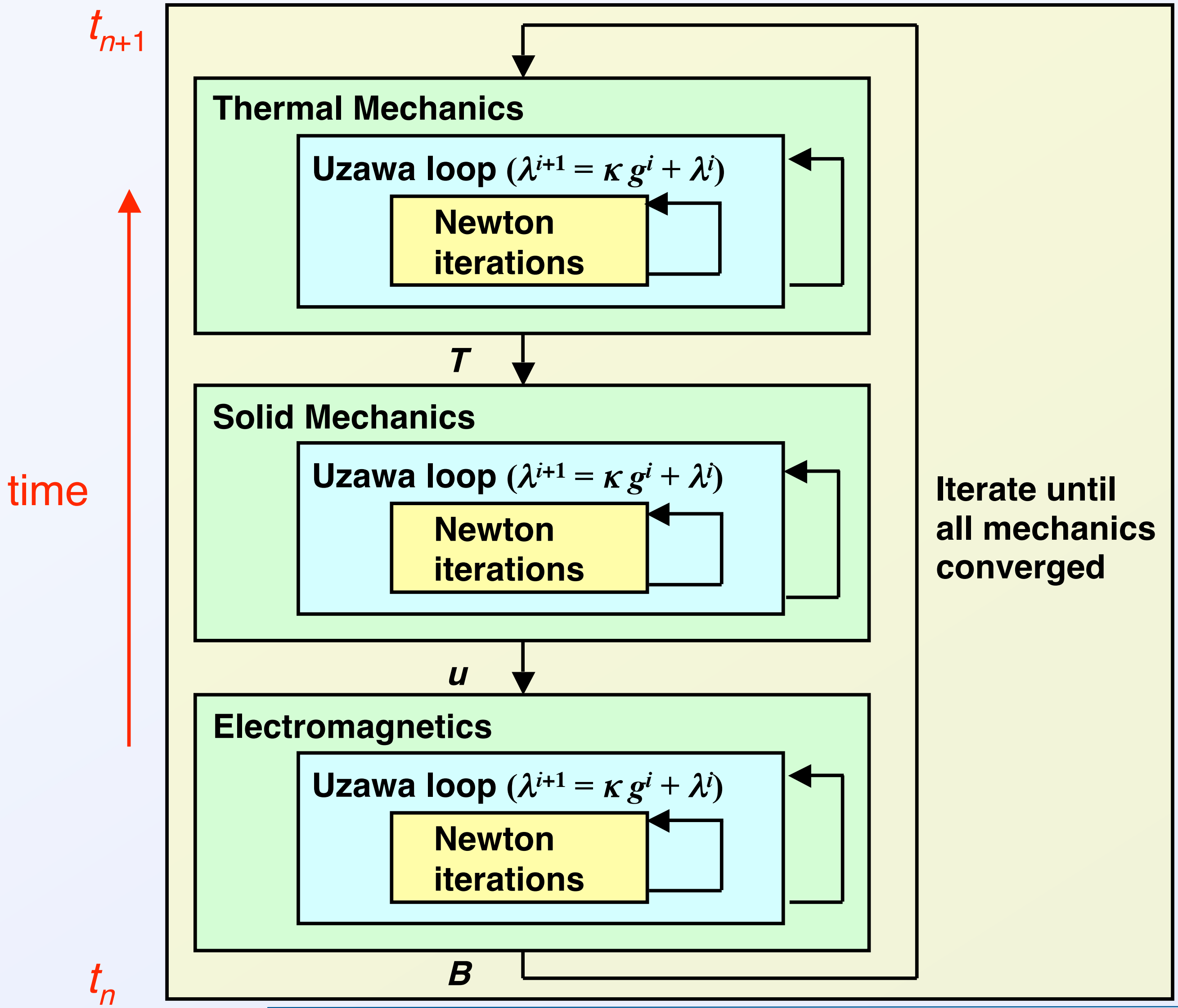
gap: $$g_A = \int_\Gamma N_A (x^s - x^m)\, d\Gamma$$

penalty: $$\lambda_A = \kappa g_A$$

augmented Lagrange: $$\lambda_A^{i+1} = \kappa g_A^i + \lambda_A^i$$

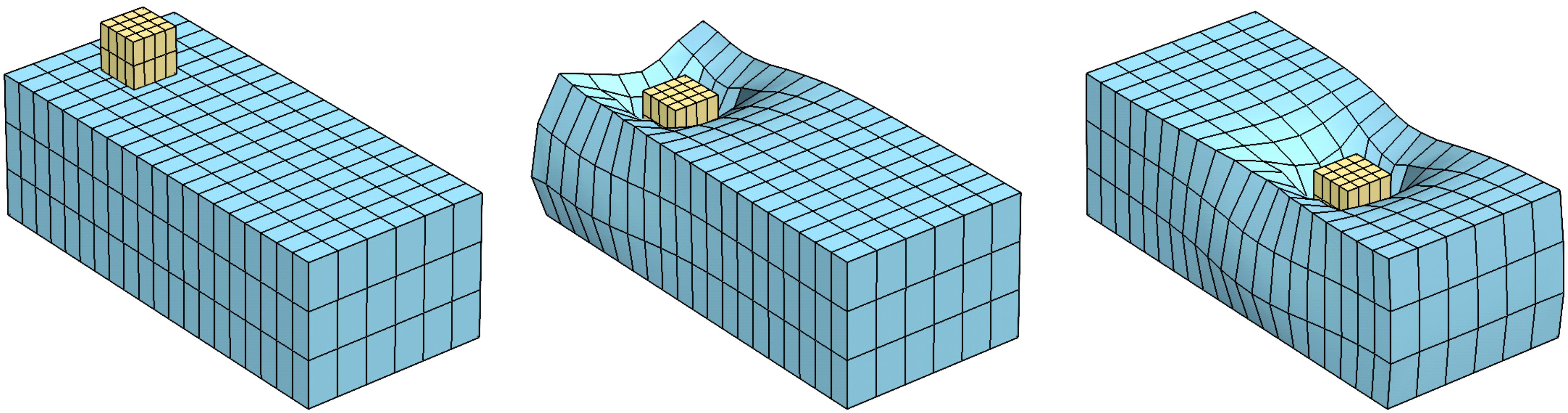


Partitioned Coupled Approach



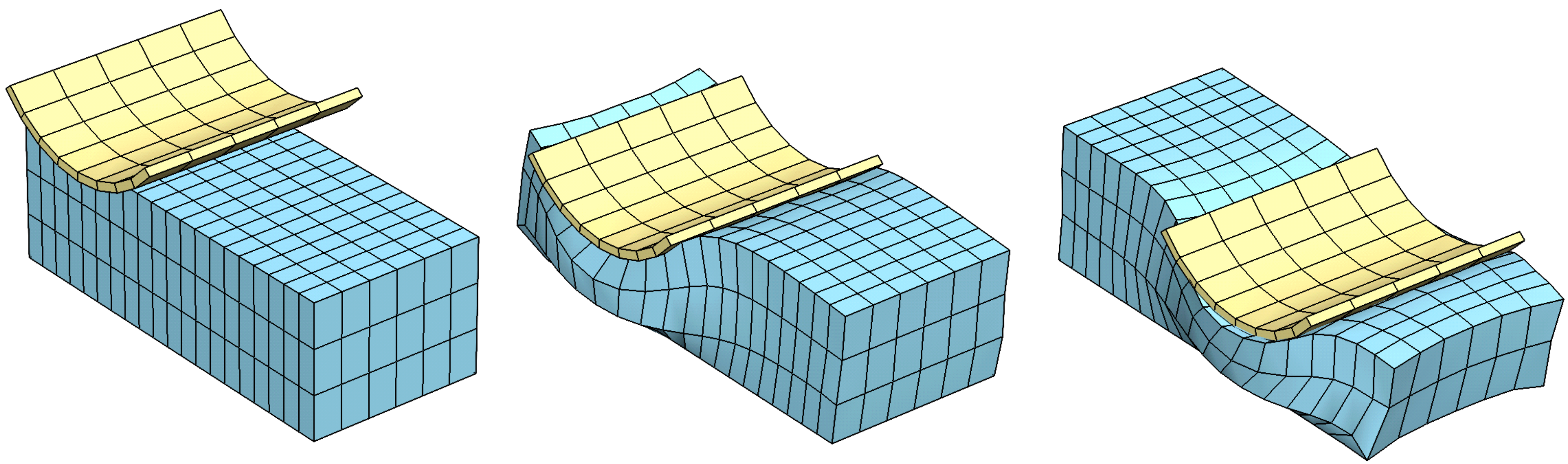
Mortar Segment-to-Segment Results

- Mortar Segment-to-Segment solves implicit problems Node-on-Segment can't



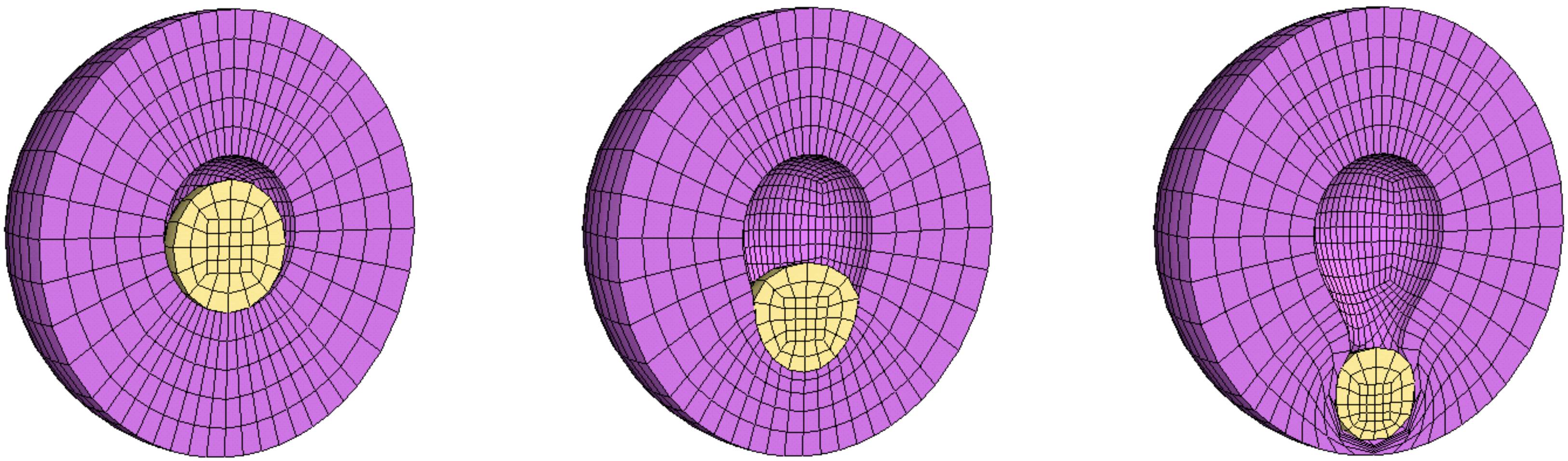
Mortar Segment-to-Segment Results

- Mortar Segment-to-Segment solves implicit problems Node-on-Segment can't



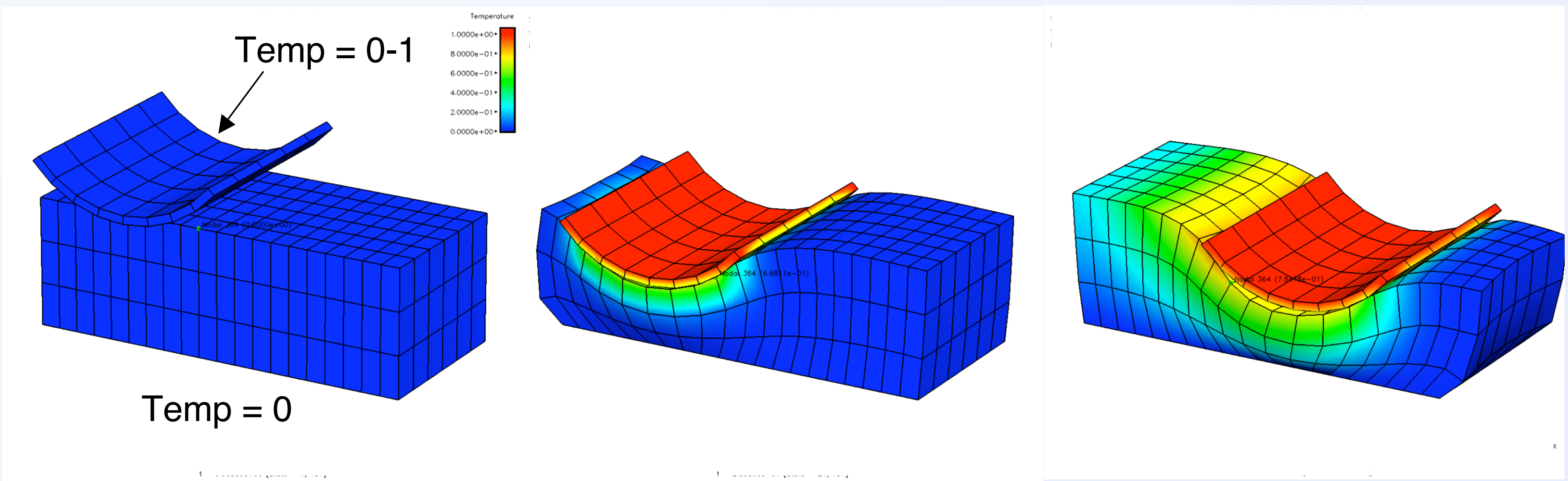
Mortar Segment-to-Segment Results

- Mortar Segment-to-Segment solves implicit problems Node-on-Segment can't



Mortar Segment-to-Segment Results

- Mortar Segment-to-Segment with coupled solid-thermal

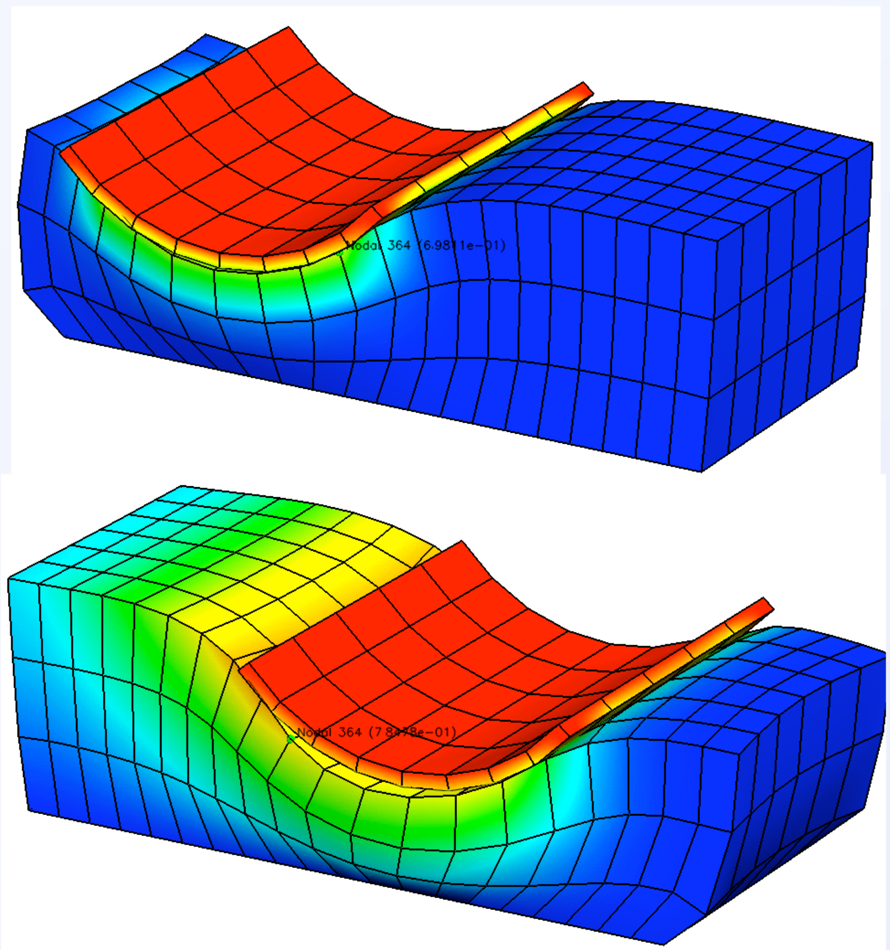


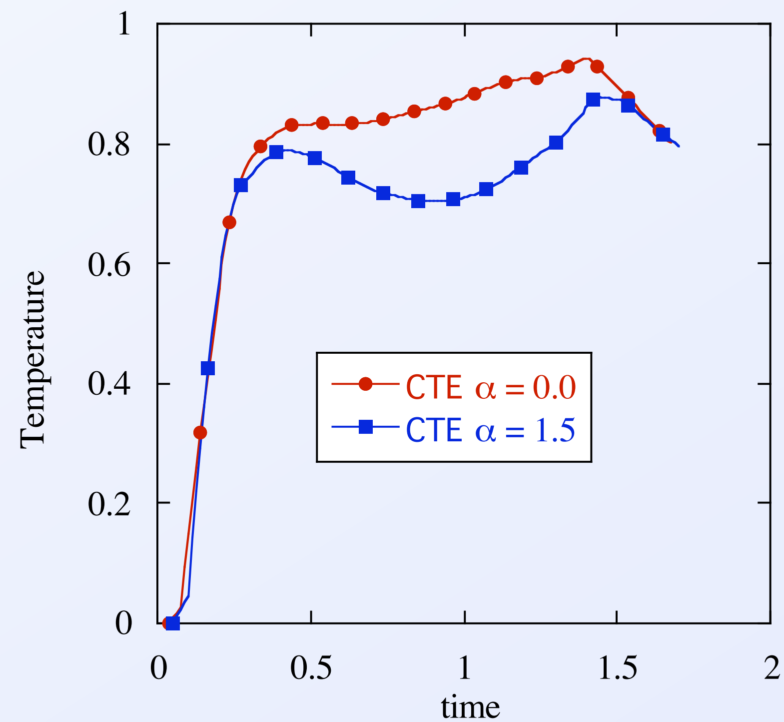
<u>Material Parameters</u>:

Solid:	Young's modulus	E = 1.0
	Poisson's Ratio	$\nu = 0.3$
	CTE	$\alpha = 0.0$ and 1.5
Thermal:	Thermal Conductivity	k = 0.375
	Heat Capacity	c = 0.461

Mortar Segment-to-Segment Results

- Mortar Segment-to-Segment with coupled solid-thermal





Note: Significant coupling of thermal and solid
$\alpha = 0.0 \Rightarrow e_{tol} = 1.0E\text{-}13$ or better
$\alpha = 1.5 \Rightarrow e_{tol} = 1.0E\text{-}05$ (okay)
Conclusion: partitioned solution method not robust

Electromagnetics

- Applications: Rail Gun, Flux Compression Generator etc.
  - Equations:

$$\int_{\Gamma} \mathbf{J} \cdot d\mathbf{\Gamma} = \int_{\partial\Gamma} \frac{1}{\mu} \mathbf{B} \cdot d\mathbf{l} \qquad \text{Ampere's law}$$

$$\frac{d}{dt} \int_{\Gamma(t)} \mathbf{B} \cdot d\mathbf{\Gamma} = - \int_{\partial\Gamma} \mathbf{E}' \cdot d\mathbf{l} \qquad \text{Faraday's law}$$

$$\mathbf{J} = \mathbf{f}(\mathbf{E}_{\mathrm{Mat}}) \qquad \text{Constitutive law for current}$$

$$\nabla \cdot \mathbf{B} = 0 \qquad \text{Conservation of magnetic flux}$$

$$\nabla \cdot \mathbf{J} = 0 \qquad \text{Conservation of current}$$

  - Along interface

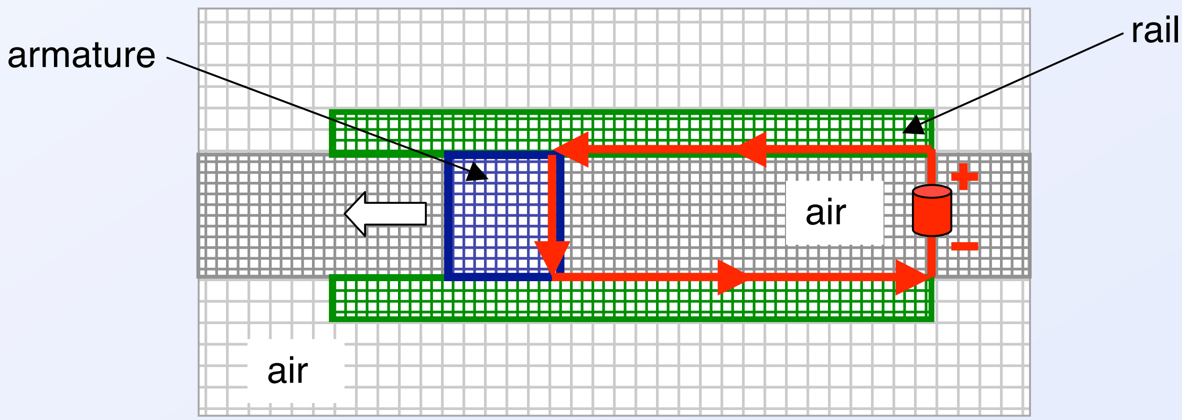
$$[\![\mathbf{E}_{\mathrm{Mat}}]\!] \times \mathbf{n} = \left([\![\mathbf{v}]\!] \times B_n \, \mathbf{n} \right) \times \mathbf{n} \,,$$

$$[\![\mathbf{B}]\!] \cdot \mathbf{n} = 0 \,,$$

$$[\![\mathbf{J}]\!] \cdot \mathbf{n} = 0 \,.$$

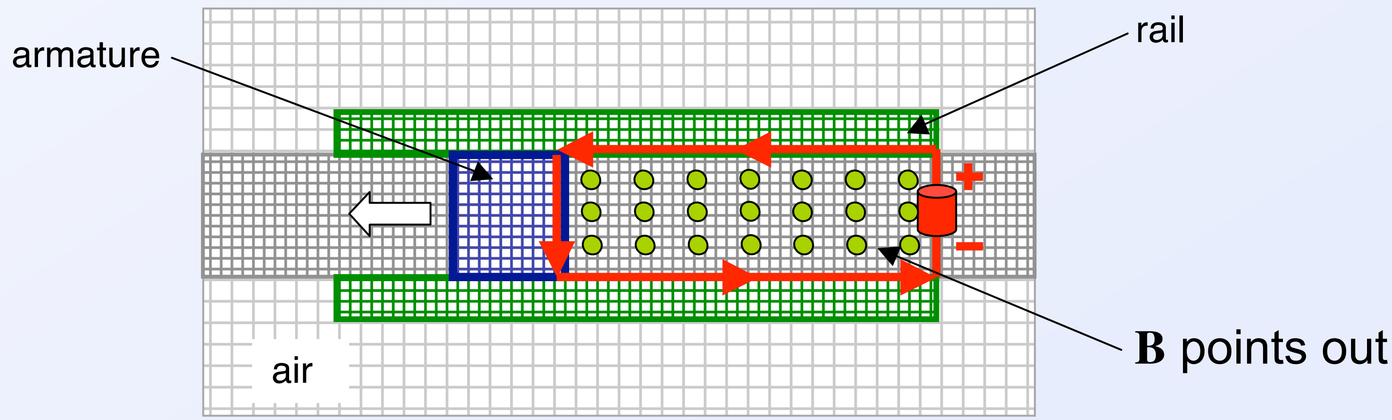
Electromagnetics: Rail Gun

- Coupled solid mechanics - electromagnetics used to simulate Rail Gun
  - First of a kind implementation of electromagnetic sliding contact
    - Used mortar method
  - Amperes Law: $\nabla \times \mathbf{B} = \mu \mathbf{J}$ since $\mu_{air} >> \mu_{iron}$
    - $\mathbf{B}$ field is small inside iron armature
    - $\mathbf{B}$ field is high in air next armature armature
  - Lorentz Force: $\mathbf{F} = \mathbf{J} \times \mathbf{B}$ causes net force on armature



Electromagnetics: Rail Gun

- Coupled solid mechanics - electromagnetics used to simulate Rail Gun
  - First of a kind implementation of electromagnetic sliding contact
    - Used mortar method
  - Amperes Law: $\nabla \times \mathbf{B} = \mu \mathbf{J}$ since $\mu_{air} >> \mu_{iron}$
    - $\mathbf{B}$ field is small inside iron armature
    - $\mathbf{B}$ field is high in air next armature armature
  - Lorentz Force: $\mathbf{F} = \mathbf{J} \times \mathbf{B}$ causes net force on armature

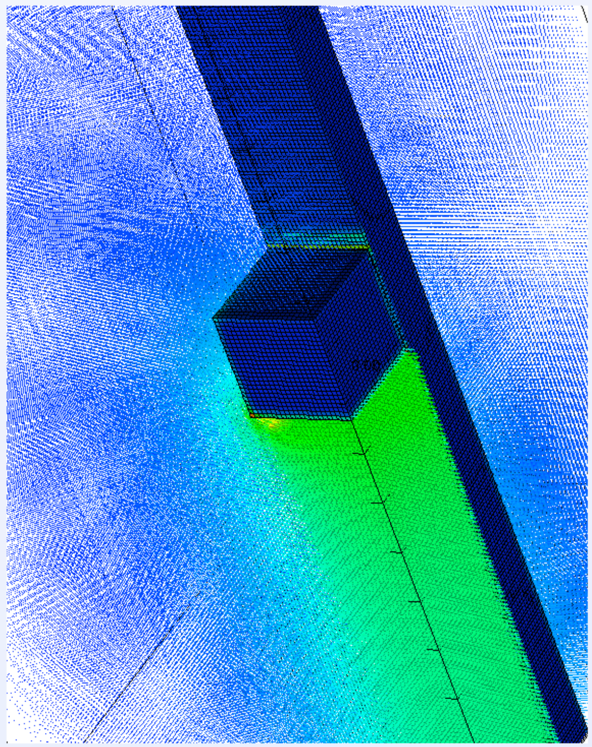


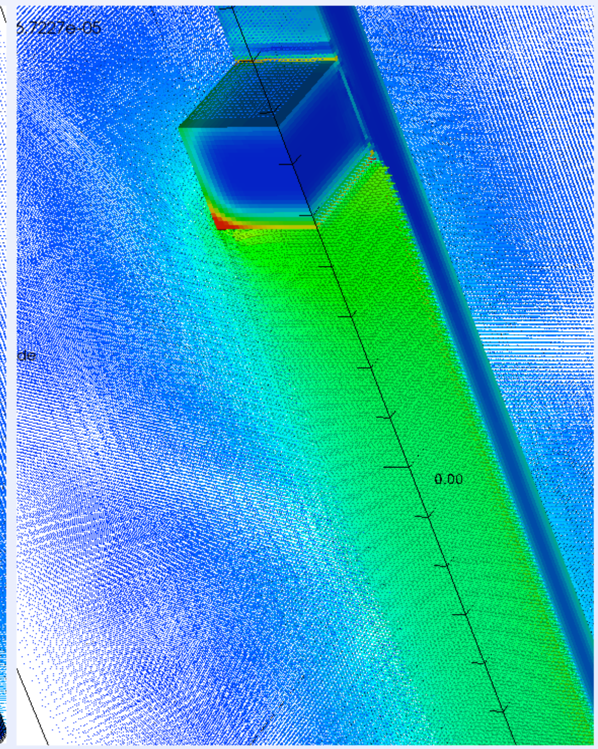
Electromagnetics: Rail Gun

- Try to answer why Rail Guns don’t reach theoretical speed
- Currently comparing to experimental tests
- Plan to add plasma model. Plasma trails armature and carries current.
- Algebraic multigrid solver currently doesn’t work with contact: use PWSMP

<u>Solution Data</u>

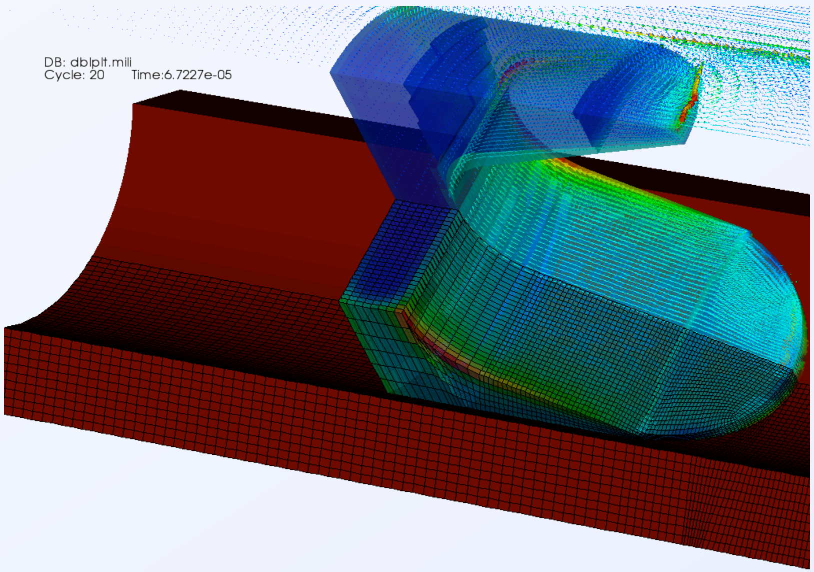
– 1/4 Symmetry
– 1.8 million elements
– 20 time steps ($3x10^{-6}$s)
– 15 hours to solve (Atlas)
– 32 partitions
– 4 threads per partition
– 128 processors total

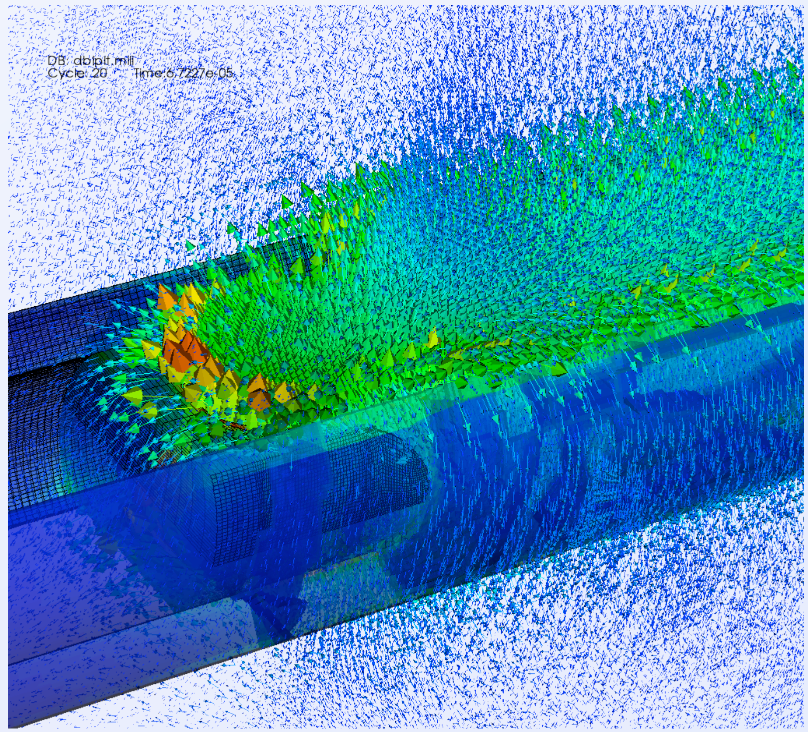




Electromagnetics: Rail Gun

- “Horseshoe” design

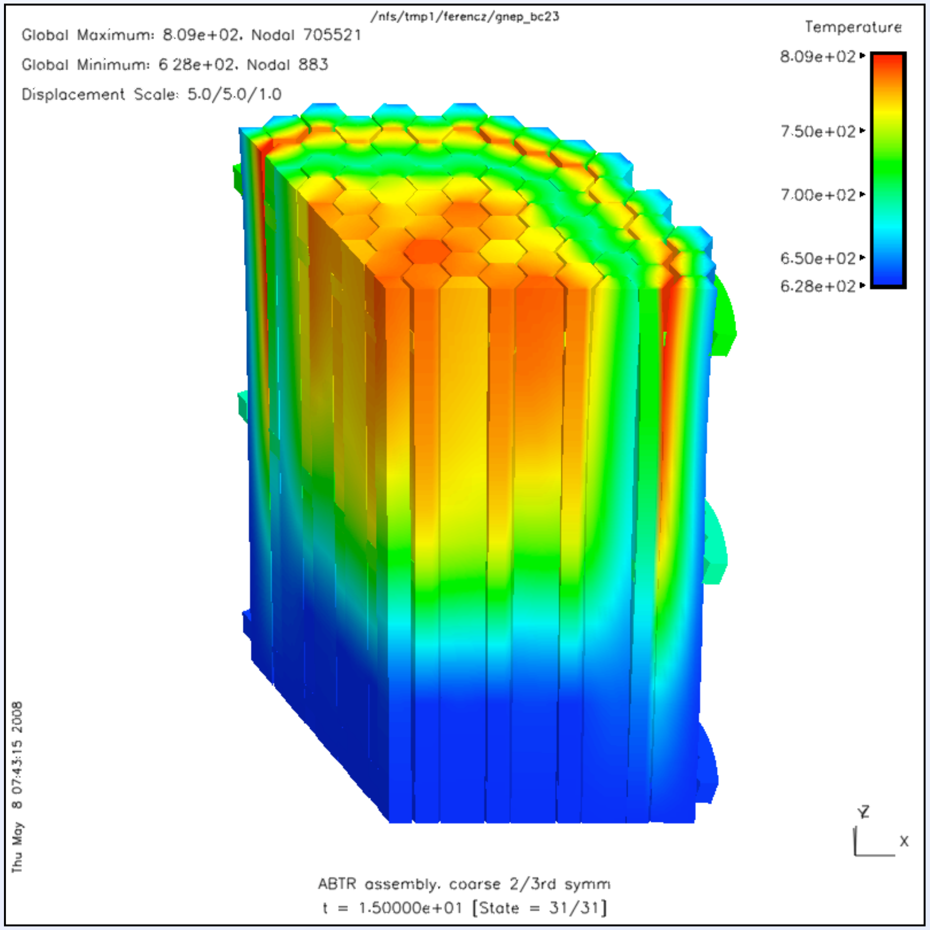




Nuclear Reactor

- Coupled solid-thermal mechanics

Solution Data
- 1/3 Symmetry
- 75 Hexagonal Rods held by 3 restraint plates
- 390 mortar contact surfaces
- 2 million dofs
- 32 partitions, 4 threads/partition
- 128 processors total
- Neutronics package hands off heat sources to thermo-mech code
- Thermal gradient causes rod bending and open gaps: petaling

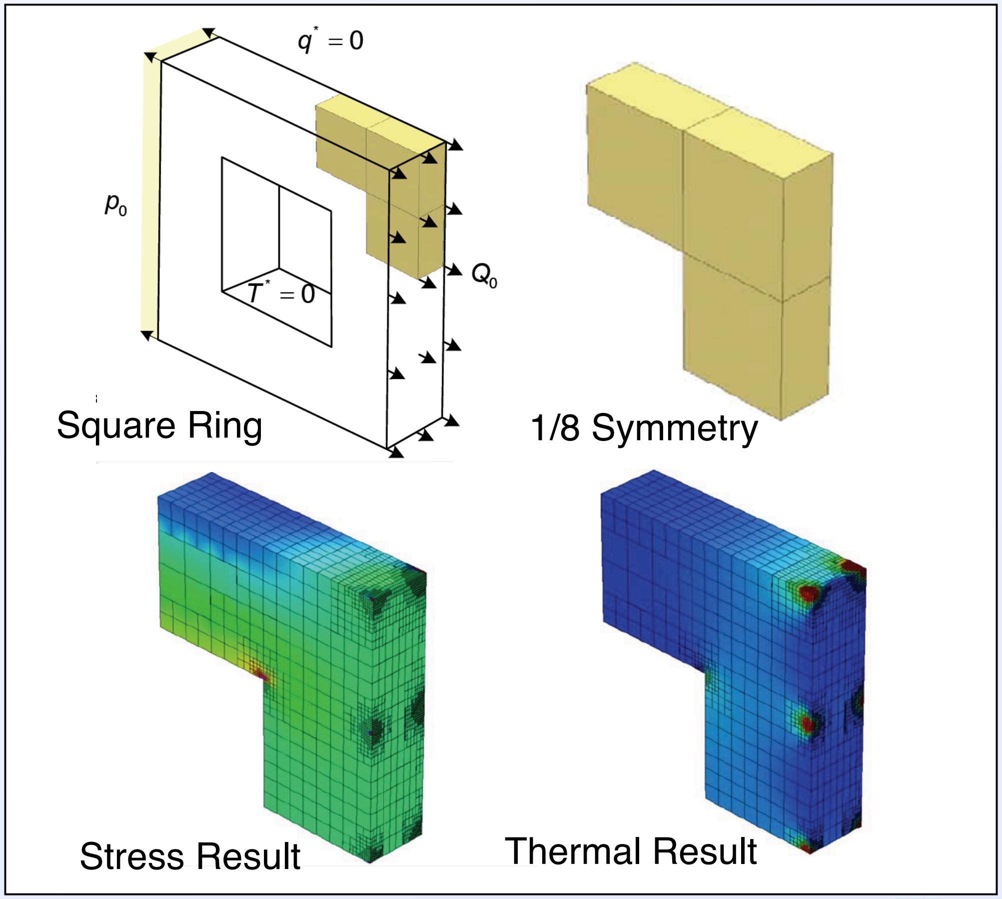


Adaptivity

- Currently use just one mesh for all mechanics
  - Consider thermal-mechanical problem

- 1/8 symmetry of square ring
- Solid pressure b.c.
- Thermal point sources

Thermal and stress concentrations not always in the same spot

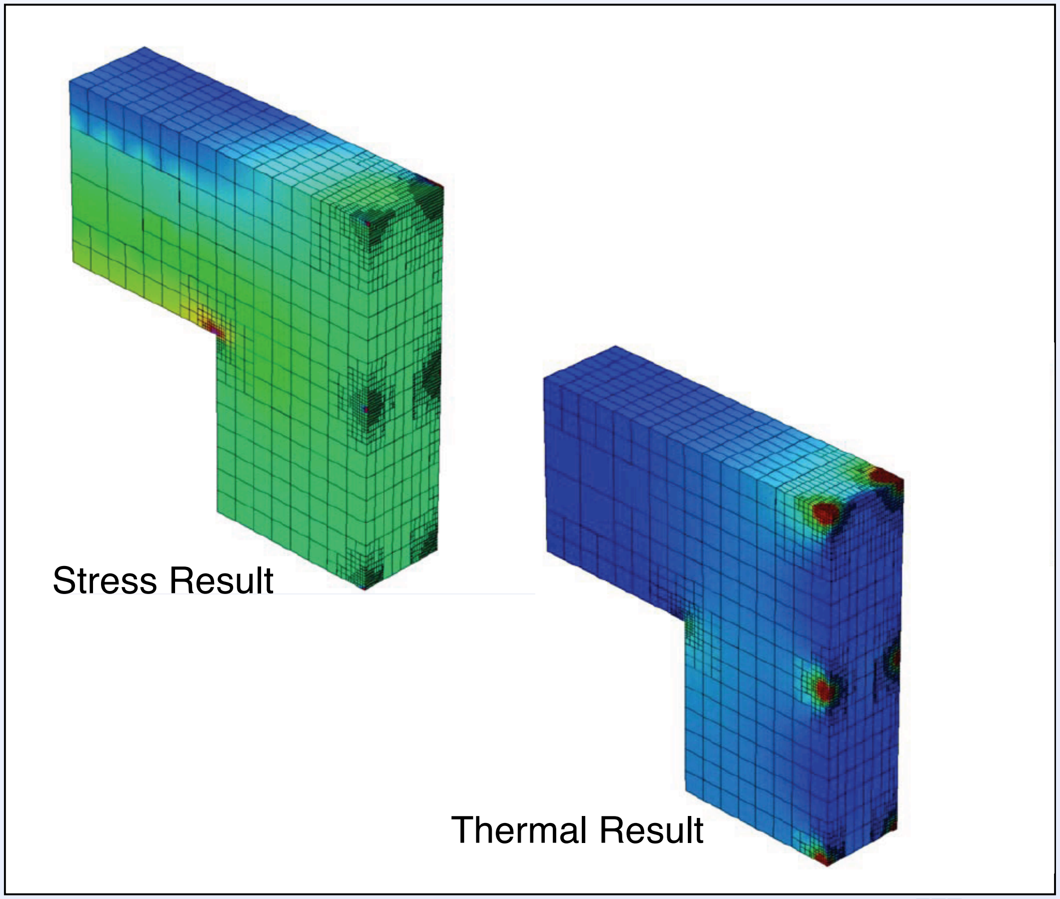


Adaptivity

- Currently use just one mesh for all mechanics
  - Consider thermal-mechanical problem

- 1/8 symmetry of square ring
- Solid pressure b.c.
- Thermal point sources

Thermal and stress concentrations not always in the same spot



Work in Progress

- Couple mechanics using *different* meshes
  - Exploit mesh mapping program
    - How to move mesh?
- Incorporate tight coupling of neutronics and hydraulics into thermal mechanics code.
  - Neutron transport affected by geometry
  - Hydraulics models coolant
- Modify algebraic multi-grid solver to accommodate contact
- Extend embedded mesh technology to treat coupled solid-electromagnetics model of rail gun
- Add “cracks” in ship hulls
  - XFEM joint work with Northwestern Univ. and Indian Head Naval Base